

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 22, 2013

J. Arango
S. Venaas
I. Kouvelas
Cisco Systems
February 18, 2013

PIM Join Attributes for LISP Environments
draft-arango-pim-join-attributes-for-lisp-00.txt

Abstract

This document defines two PIM Join/Prune attributes that support the construction of multicast distribution trees where the root and receivers are located in different LISP sites. These attributes allow the receiver site to select between unicast and multicast transport and to convey the receiver RLOC address to the control plane of the root xTR.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 22, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Requirements Notation	4
3. PIM Join/Prune Attributes	5
4. The Transport Attribute	6
4.1. Transport Attribute Format	6
4.2. Using the Transport Attribute	6
5. Receiver RLOC Attribute	8
5.1. Receiver RLOC Attribute Format	8
5.2. Using the Receiver RLOC Attribute	9
6. Security Considerations	10
7. IANA Considerations	11
8. Normative References	12
Authors' Addresses	13

1. Introduction

The construction of multicast distribution trees where the root and receivers are located in different LISP sites [RFC6830] is defined in [RFC6831]. Creation of (root-EID,G) state in the root site requires that unicast LISP-encapsulated Join/Prune messages be sent from an xTR on the receiver site to an xTR on the root site.

[RFC6831] specifies that (root-EID,G) data packets are to be LISP-encapsulated into (root-RLOC,G) multicast packets. However, a wide deployment of multicast connectivity between LISP sites is unlikely to happen any time soon. In fact, some implementations are initially focusing on unicast transport with head-end replication between root and receiver sites.

The unicast LISP-encapsulated Join/Prune message specifies the (root-EID,G) state that needs to be established in the root site, but conveys nothing about the receivers capability or desire to use multicast as the underlying transport. This document specifies a Join/Prune attribute that allows the receiver to select the desired transport.

Knowledge of the receiver RLOC is also essential to the control plane of the root xTR. It determines the downstream destination for unicast head-end replication and identifies the receiver xTR that needs to be notified should the root of the distribution tree move to another site.

The outer source address field of the encapsulated Join/Prune message contains an RLOC address of the receiver xTR. This source address is message to the root xTR RLOC destination. Due to policy and load balancing considerations, the selected source address may not be the RLOC on which the receiver site wishes to receive a particular flow. This document specifies a Join/Prune attribute that conveys the appropriate receiver RLOC address to the control plane of the root xTR.

2. Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. PIM Join/Prune Attributes

PIM Join/Prune attributes are defined in [RFC5384] by introducing a new Encoded-Source type that, in addition to the Join/Prune source, can carry multiple type-length-value (TLV) attributes. These attributes apply to the individual Join/Prune sources on which they are stored.

The attributes defined in this document conform to the format of the encoding type defined in [RFC5384]. The attributes would typically be the same for all the sources in the Join/Prune message. Hence we RECOMMEND using the hierarchical Join/Prune attribute scheme defined in [I-D.venaas-pim-hierarchicaljoinattr]. This hierarchical system allows attributes to be conveyed on the Upstream Neighbor Address field, thus enabling the efficient application of a single attribute instance to all the sources in the Join/Prune message.

LISP xTRs do not exchange PIM Hello Messages and hence no Hello option is defined to negotiate support for these attributes. Systems that support unicast head-end replication are assumed to support these attributes.

4. The Transport Attribute

It is essential that a mechanism be provided by which the desired transport can be conveyed by receiver sites. Root sites with multicast connectivity will want to leverage multicast replication. However, not all receiver sites can be expected to have multicast connectivity. It is thus desirable that root sites be prepared to support (root-EID,G) state with a mixture of multicast and unicast output state. This document specifies a Join/Prune attribute that allows the receiver to select the desired underlying transport.

4.1. Transport Attribute Format

```

      0                               1                               2
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |F|E| Type = 5 | Length = 1 | Transport |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

F-bit: The Transitive bit. Specifies whether the attribute is transitive or non-transitive. MUST be set to zero. This attribute is ALWAYS non-transitive.

E-bit: End-of-Attributes bit. Specifies whether this attribute is the last. Set to zero if there are more attributes. Set to 1 if this is the last attribute.

Type: The Transport Attribute type is 5.

Length: The length of the Transport Attribute value. MUST be set to 1.

Transport: The type of transport being requested. Set to 0 for multicast. Set to 1 for unicast.

4.2. Using the Transport Attribute

Hierarchical Join/Prune attribute instances [I-D.venaas-pim-hierarchicaljoinattr] SHOULD be used when the same Transport Attribute is to be applied to all the sources within the Join/Prune message or all the sources within a group set. The root xTR MUST accept Transport Attributes in the Upstream Neighbor Encoded-Unicast address, Encoded-Group addresses, and Encoded-Source addresses.

There MUST NOT be more than one Transport Attribute within the same encoded address. If an encoded address has more than one instance of the attribute, the root xTR MUST discard all affected Join/Prune sources.

5. Receiver RLOC Attribute

The root xTR must know the receiver RLOC addresses of all receiver sites for a given (root-EID,G) so that it can perform unicast LISP-encapsulation of multicast data packets to each and every receiver site that has requested unicast head-end replication.

To support mobility of EIDs, the root xTR must keep track of ALL receiver RLOCs even when the corresponding downstream site has not requested unicast replication. The root xTR may detect that a local multicast source "root-EID" has moved to a remote LISP site. Under such circumstances LISP sends a SMR message to all receiver xTRs, prompting them to update their map cache. This is only possible if LISP can obtain from PIM the set of all receiver RLOCs that have active Join state for the root-EID.

The outer source address field of the encapsulated Join/Prune message contains an RLOC address of the receiver xTR. LISP xTRs, as edge devices, are commonly subject to URPF checks by the network providers on each core-facing interface. The source address for the encapsulation header must therefore be the RLOC of the core-facing interface used to physically transmit the encapsulated Join/Prune message. Due to policy and load balancing considerations, that may not be the RLOC on which the receiver site wishes to receive a particular flow. This document specifies a Join/Prune attribute that conveys the appropriate receiver RLOC address to the control plane of the root xTR.

5.1. Receiver RLOC Attribute Format

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|F|E| Type = 6 | Length | Addr Family | Receiver RLOC
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+...

```

F-bit: The Transitive bit. Specifies whether this attribute is transitive or non-transitive. MUST be set to zero. This attribute is ALWAYS non-transitive.

E-bit: End-of-Attributes bit. Specifies whether this attribute is the last. Set to zero if there are more attributes. Set to 1 if this is the last attribute.

Type: The Receiver RLOC Attribute type is 6.

Length: The length in octets of the attribute value. MUST be set to the length in octets of the receiver RLOC address plus one octet to account for the Address Family field.

Addr Family: The PIM Address Family of the receiver RLOC as defined in [RFC4601].

Receiver RLOC: The RLOC address on which the receiver xTR wishes to receive the unicast-encapsulated flow.">

5.2. Using the Receiver RLOC Attribute

Hierarchical Join/Prune attribute instances [I-D.venaas-pim-hierarchicaljoinattr] SHOULD be used when the same Receiver RLOC attribute is to be applied to all the sources within the message or all the sources within a group set. The root xTR MUST accept Transport Attributes in the Upstream Neighbor Encoded-Unicast address, Encoded-Group addresses, and Encoded-Source addresses.

There MUST NOT be more than one Receiver RLOC Attribute within the same encoded address. If an encoded address has more than one instance of the attribute, the root xTR MUST discard all affected Join/Prune sources.

6. Security Considerations

Security of the Join Attribute is only guaranteed by the security of the PIM packet. The attributes specified herein do not enhance or diminish the privacy or authenticity of a Join/Prune message. A site that legitimately or maliciously sends and delivers a Join/Prune message to another site will equally be able to append these and any other attributes it wishes.

7. IANA Considerations

Two new PIM Join/Prune attribute types need to be assigned. Type 5 is being requested for the Transport Attribute. Type 6 is being requested for the Receiver RLOC Attribute.

8. Normative References

- [AFI] IANA, "Address Family Numbers",
<http://www.iana.org/assignments/address-family-numbers>.
- [I-D.venaas-pim-hierarchicaljoinattr]
Venaas, S., Kouvelas, I., and J. Arango, "Hierarchical Join/Prune Attributes",
draft-venaas-pim-hierarchicaljoinattr-00 (work in progress), February 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.
- [RFC5384] Boers, A., Wijnands, I., and E. Rosen, "The Protocol Independent Multicast (PIM) Join Attribute Format", RFC 5384, November 2008.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.

Authors' Addresses

Jesus Arango
Cisco Systems
170 Tasman Drive
San Jose, CA 95134
USA

Email: jeearango@cisco.com

Stig Venaas
Cisco Systems
170 Tasman Drive
San Jose, CA 95134
USA

Email: stig@cisco.com

Isidor Kouvelas
Cisco Systems
170 Tasman Drive
San Jose, CA 95134
USA

Email: kouvelas@cisco.com

LISP Working Group
Internet-Draft
Intended status: Experimental
Expires: January 03, 2014

S. Barkai
ConteXtream Inc.
D. Farinacci

D. Meyer

F. Maino
V. Ermagan
Cisco Systems
July 02, 2013

LISP Based FlowMapping for Scaling NFV
draft-barkai-lisp-nfv-02

Abstract

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV). Network functions such as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware appliances embedded into the network as turn-key service-nodes or service-blades within routers. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled virtualized components of capacity and functionality. LISP-SDN based flow-mapping, dynamically assembles these components to whole solutions by steering the right traffic in the right sequence to the right virtual function instance.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 03, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Connectivity Model	4
4. Flow-Mapping Elements	7
5. Day-in-life of a Mapped Flow	8
5.1. XTR Flow Edge	9
5.2. Map Resolvers-Servers	11
5.3. XTRs-Mappers Scaling	11
6. Message Formats	11
7. QOS and Echo Measurements	14
8. Security Considerations	14
9. IANA Considerations	14
10. Acknowledgements	14
11. Normative References	14
Authors' Addresses	14

1. Introduction

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV). [RFC6830] Network functions such as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware appliances embedded into the network as turn-key service-nodes or service-blades within routers.

This monolithic service delivery method increases the complexity of service roll-out and capacity planning, limits providers' choices, and slows down revenue generating service innovation. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled ("googlized") virtualized components of capacity and functionality. Such a component based model opens up service provider networks to the savings of elasticity and open architecture driven innovation. However this model also presents the network with the new challenges of assembling components, developed by 3rd parties, into whole solutions, by forwarding the right traffic to the right function-block at the right sequence.

While this is possible, to some extent, by traditional virtual networking - virtual bridges(vBridges) and virtual-routing-forwarding (VRF) - these mechanisms are relatively static and require complex and intensive configuration of network interfaces, while elastic components are not network topology bound. Software-defined-networks, (SDN) flow based models are much more dynamically programmable but are also very centralized and hence have limited scale and resiliency. By enhancing SDN models with RFC6830 overlay model, as this specification suggests, we offer a best fit to dynamic assembly of virtualized network functions in the service-providers data-centers and distribution-centers.

2. Terminology

The following terms are used to describe a LISP based implementation of Software-Defined Flow-Mapping-Fabric for NFV:

- o LISP-SDN - is an enhancement to the basic SDN model of (1) hop-to-hop (2) push-down flow-commands (3) by concentrated-controller.. to a LISP based architecture of (1) distributed-overlay e.g. SDN over IP (2) based on a pull-publish-subscribe actions from xTR-edges up.. (3) to a global mapping service. A mapping service scaled by and connected over the IP underlay network.
- o Virtualized Network Function (VNF) - is a process instance with an EID and RLOC that performs a defined set of inline network functions. a VNF can be software on a virtual-machine (VM) performing a function like multimedia signaling, mobility management, content caching or streaming, security, filtering, optimization, etc. A VNF class type and VNF instance capacity, load, and location are attributes that can be resolved by the LISP-SDN mapping service.
- o Client-Flow - is a sequence of packets that corresponds to a specific communication thread or network conversation between a

client application and a network service. Client-flows are typically processed by various in-network functions either as the end service side to the network conversation, or as middle-box functionality.

- o SDN-xTR - is a LISP xTR element that classifies traffic into application flows, maps, encapsulates, and decapsulates flows in order to emerge a flow-mapping solution - along with a collection of the SDN-xTR elements, and the LISP-SDN mapping service.
- o SDN-Overlay - is the network formed by the collection of inter-connected SDN-xTR
- o SDN-Underlay - is the IPvN network connecting SDN-xTRs
- o SDN-Outerlay (interim name)- is the collection of networks and interfaces aggregated by the various SDN-xTRs connecting VNFs and Client-flows coming from access networks or the Internet.
- o Flow-Rule - is a set of pattern tuples that match any part of a packet header and is used to classify packets into flows as well as trigger forwarding actions such as encapsulation / decapsulation, network address translation (NAT), etc. We differentiate between exact-match rules (many) which include an exact set of tuple bits, and best match rules (fewer) which contain both tuple bits and wild-cards "*".
- o Virtual IP (VIP) - is an IP address or EID that identifies a function rather than a specific destination. For example all the encapsulated client-flow traffic sent from a base-station eNodeBs over a transport network, can have as destination a VIP which represents in a given LISP-SDN solution, the function mobile-gateway or PGW, and not any specific destination.
- o Flow-Affinity - is the association between a client-flow and a VNF instance. VNF logic will typically create long-lived (minutes) in memory states in order to perform its functions. Therefore once an affinity is established it is best to keep it for as long as possible in order not to stress or break the VNF application.

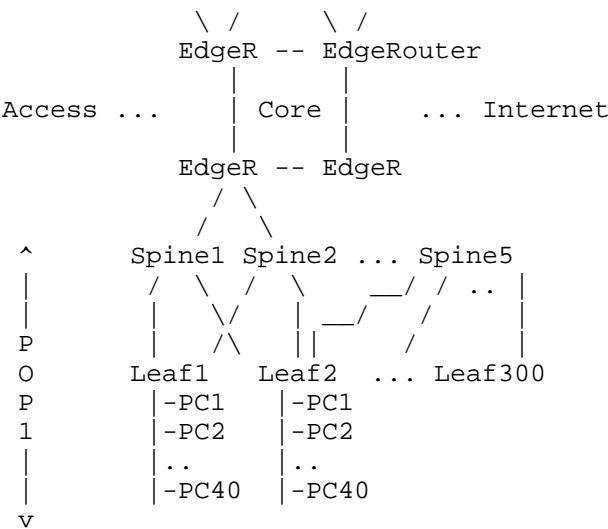
3. Connectivity Model

The basic connectivity model used to assemble VNFs into whole solution is the flow-mapping-fabric. Unlike topological forwarding which is based on source-subnet >> routed hop by hop >> destination-subnet, a flow-mapping-fabric maps, forwards and "patches" flows by identity directly to the end systems. The identities used for the flow-mapping-fabric are those associated with the client-flows e.g.

Subscriber ID, phone number, TCP port, etc. and those associated with the VNF e.g. the type, location, physical address, etc. the flow-mapping-fabric is implemented as a LISP-SDN overlay, over in-place IP underlay, assembling outerlay flows into solutions. Bellow are basic assumptions regarding the Underlay, Outerlay, and Overlay in the solution:

- o The underlying physical network is assumed to be topology based and implemented using standard bridging and routing. Conventional design principles are applied in order to achieve both capacity and availability of connectivity. Typical examples of underlays include spine-leaf switching for clustering server racks, and, core-edge routing inter-connecting server clusters across points of presence. Edge networks are also used to connect to access networks and Internet.
- o The flow-mapping-fabric maps outerlay client-flows to VNFs. This enables assembly, scaling, balanced high-utilization, massive concurrency, and hence, performance of NFVs. By mapping each client-flow to the correct functional instance the system engages as many VNF components as are available, scaled within and across data-centers. Applied recursively client-flow mapping can chain a sequence of VNF components to make up an end-to-end service.
- o The overlay network is based on location-identity-separation and forms a virtualization indirection ring around spines and cores. The overlay edges aggregate outerlay client-flows and VNFs. Outerlay flows are classified, mapped, and encapsulated over the edge through the underlay interfaces and are transported to the right identity's locations.

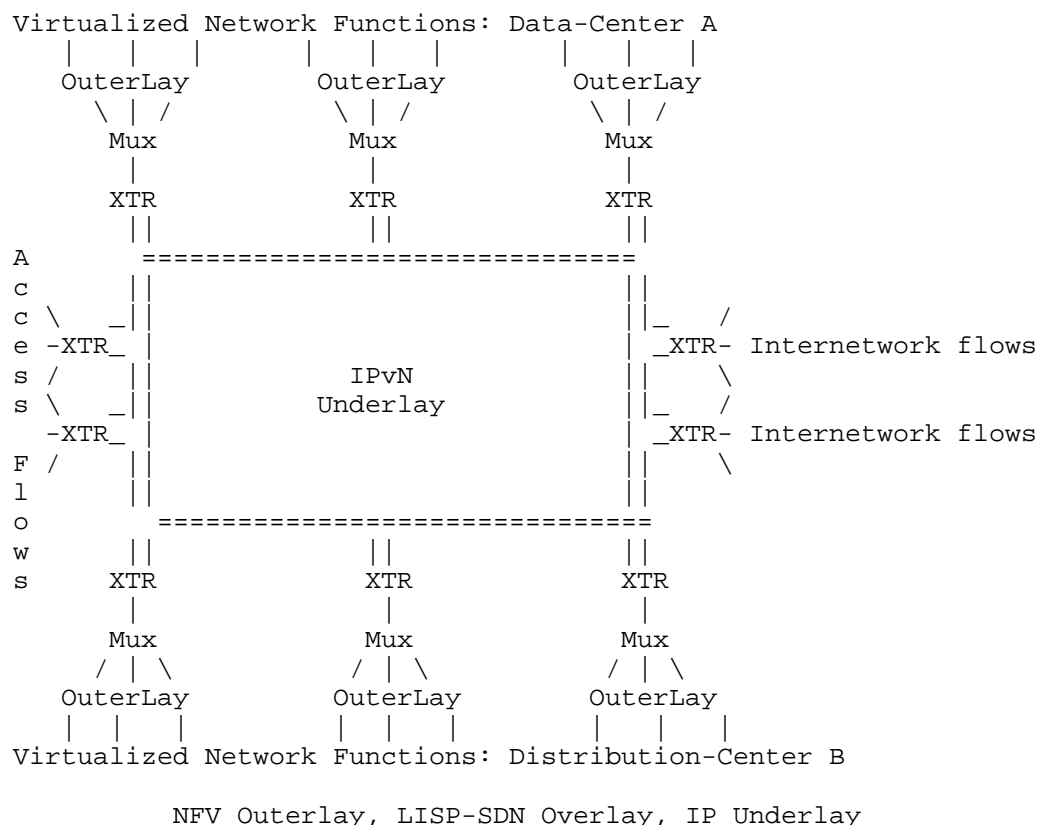
POP3 POP4



Core-Edge Spine-Leaf Underlays

	FunctionA				FunctionB				..				FunctionN							
v	v				v				v				v							
Recursion	Instancel..i				Instancel..j				Instancel..k											
v																				
v																				
SubsFlow1	o	o	o	o	-	-	+	o	o	o	o	-	-	-	o	o	o	o		
SubsFlow2	o	+	o	o	-	-	-	o	o	o	o	-	-	-	o	o	o	o		
.			
.			
.			
SubsFlowM	o	o	o	o	-	-	-	o	o	o	o	-	-	-	+	o	o	o		

Flow-Mapping-Fabric

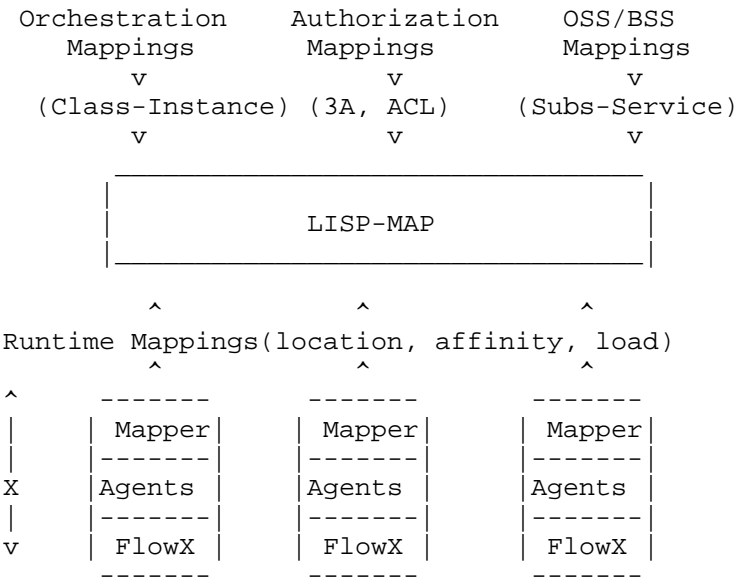


4. Flow-Mapping Elements

In order to implement NFV Flow-Mapping-Fabric using LISP-SDN We use the following components and capabilities:

1. **Flow-Switching:** is a component within an SDN-xTR and contains a set of n-tuple flow-rules matched against each packet in order to separate it to (LOCALLY defined) sequences representing flows. Flows are either Encapsulated into the Overlay, decapsulated to the Outerlay, or forwarded to SDN-xTR Control Agents.
2. **Control-Agents:** are software processes running in SDN-xTRs and are invoked for each flow where an exact match was not present in the Flow-Switching. The default "catch-all" Flow-Handler maps IP flows to locations and gateways based on RFC 6830. Protocol and application specific handlers can be loaded into the SDN-xTR for handling specific mapping and AFFINITY requirements of network functions. Examples of such protocols and applications can be SIP, GTP, SLX etc.

3. Global-Mapping: is how GLOBALLY significant key-value mappings is translated to LOCALLY defines flow masks and encapsulation actions. Examples of such mappings include: Map a functional instance ID to a function class ID; map subscriber-application ID to virtual function instance ID; map instance ID to location; instance to health, load, tenant; etc.



Identity-Location Overlay

5. Day-in-life of a Mapped Flow

Let us walk through detailed steps of the use of RFC6830 and LISP architecture in order to perform resource virtualization and flow assignment to virtual function instances.

At a high level, when a client-flow packet first arrives at a SDN-xTR on the edge of the LISP overlay, the SDN-xTR must decide on a VNF instance that is best suited to service this flow, assign this flow to the selected VNF, and encapsulate this flow to the RLOC of the selected virtual function instance.

To select the best suited VNF instance, the SDN-xTR queries the Mapping System with the extracted identity parameters, both the client and the function EIDs, and receives the list of all VNF instances that represent that Function along with their RLOC and health-load attributes. The SDN-xTR runs local algorithms on the returned set to select the best suited virtual function instance.

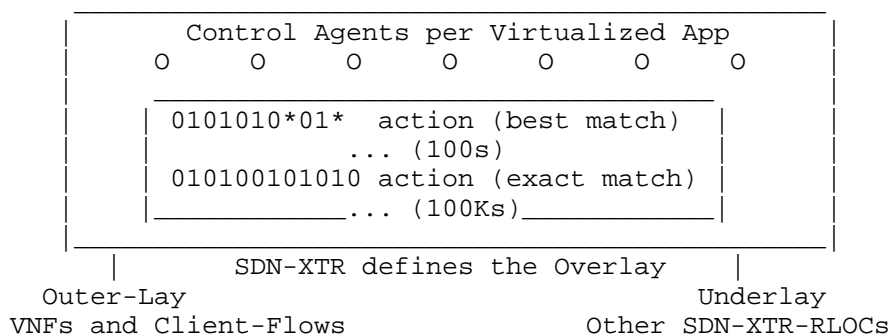
Once selected, the SDN-xTR stores (registers) the assignment of this flow to the associated VNF instance in the Mapping System. This assignment is referred to as the Affinity for this flow. The SDN-xTR also programs an exact match flow rule in its data-plane, so future packets from this flow will be mapped to the same EID-RLOC.

In the following subsections We describe this process in more detail.

5.1. XTR Flow Edge

SDN-xTR locations define the boundary of the virtual network. For the purpose of LISP-SDN flow-mapping-fabric We refer to the bellow SDN-XTR generic reference architecture. Actual vendor implementations may vary, but most likely will include similar components and structure. The SDN-XTR includes:

- o Mux-DeMux: Interfaces to the Underlay and Outerlay
- o Flow-Rules: Patterns-Actions, Exact / Best Match, Encap-Decap
- o Control-Agents: Application specific flow-handlers registered in the Flow-Rules



SDN-XTR Reference Architecture

SDN-XTR Flow Switching works as follows:

1. For traffic from the Outerlay of THIS xTR that has an exact match of all the source-dest-tags.. n-tuples, the packets are processed by rule actions including encapsulation to the RLOC of the xTR which aggregates the relevant function instance to which this flow is mapped to.

2. For traffic from the Underlay that has an exact match of all the source-dest-tags... n-tuples, the packets are processed by rule actions including decapsulation and forwarding to the Outerlay of THIS xTR.
3. Traffic from the Outer-Lay or Underlay that does NOT have an exact match of all the source-dest-tags... tuples required for normal forwarding, packets are forwarded to the control agent registered in the best-matching rule.

SDN-XTR Control Agents work as follows:

1. Mapping agent type and application scope is defined by the best match entries that point to it. Control agents will typically self-register in the flow-switch. XTR control-agents can register to an existing best-match rule, or instantiate a new one.
2. Typical rule-patterns are pattern-scoped by an agent registration, and can include: protocol or service type header indications; specific virtual IP addresses (VIP) that represent a service and not a specific destination; a specific source and wild-card destination; or vice versa.
3. Mapping agents work with the LISP-SDN mapping service in order to establish a global context and local considerations for mapping decision. The goal of the agents' decision is ultimately to provision the correct exact-match rule and actions that will offload the flow-packets to flow-switching described above.

The SDN-xTR control agents query the LISP-SDN Mapping System with the flow attributes including the destination VIP, as follows:

Mapping System Lookup: Map-Request (Client identity, Function-EID)

Two outcomes are possible based on whether an affinity already exists for this flow (flow has already been assigned to a virtual function instance):

o Outcome A:

- * If an affinity already exists in the Mapping System, the Mapping System returns the locator address (RLOC) associated with the Function-Instance-EID that the (Client-EID, Function-EID) is mapped to.
- * Map-Reply: ((Client-EID, Function-EID) -> Function-Instance-RLOC)

- * In this case the Mapping System also subscribes the SDN-xTR to the Function-Instance-EID, and to the (Client-EID, Function-EID) flow in order to receive updates in case of changes on these entries. Examples of these changes are change of RLOC for the Function-Instance-EID (specially if this is a virtual application), or change of affinity for (Client-EID, Function-EID) to another Function-Instance-EID.
 - * After receiving the Map-Reply from the Mapping System, the SDN-xTR programs an exact match for the flow in the xTR data-plane.
- o Outcome B:
- * If there is no affinity previously stored, the Mapping System returns a list of Records, including one Record per each instance of the Function-EID, with their associated RLOCs and flags (weight, priority).
 - * Map-Reply: (client EID, Function-Instance-Record 1, Function-Instance-Record 2...)
 - * the SDN-xTR then selects the best suited Function-Instance-EID for this flow based on local algorithms, and registers the affinity in the Mapping System. The Mapping System stores the affinity and subscribes the SDN-xTR to the affinity and to the Function-Instance-EID in the affinity, so that SDN-xTR would receive updates if any of these changes.
 - * Map-Register ((Client-EID, Function-EID) -> Function-Instance-EID)
- o Note: An SDN-xTR must be able to query for the list of App-Instance-Records even if an affinity already exists. For this purpose a flag is required in the Map-Request to indicate whether xTR wants this info or not. We can overload the M bit in Map-Request, or allocate a new bit for this.

5.2. Map Resolvers-Servers

5.3. XTRs-Mappers Scaling

6. Message Formats

This section specifies the packet formats used throughout the flow-mapping process explained above.

A Map-Request is used with a 2-Tuple Src/Dst LCAF to query the Mapping System for the affinity or list of virtual function instance records for this flow.

0										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								
Type=1 A M P S p s										Reserved										IRC										Record Count									
										Nonce . . .																													
										. . . Nonce																													
										Source-EID-AFI										Source EID Address ...																			
										ITR-RLOC-AFI 1										ITR-RLOC Address 1 ...																			
										Reserved										EID mask-len										EID-prefix-AFI = 16387									
Rsvd1										Flags										Type = 12										Rsvd2									
4 + n																				Reserved																			
Source-ML										Dest-ML										AFI = x																			
										Source-Prefix ...																													
										AFI = x										Destination-Prefix ...																			

Where:

Source-Prefix = Client-EID

Destination-Prefix = App-EID

LISP Map-Request with 2-Tuple Src/Dst LCAF

In order to specify a 5 tuple flow, rather than just a two tuple source and destination, the combination of LCAF type 12 and LCAF type 4 must be used.

If an affinity exists in the Mapping System, meaning that the flow is already assigned to a virtual function instance, then the RLOC of that Function-Instance must be returned by the Mapping System. A Map-Reply with a 2-Tuple Src/Dst Lcaf can be used for this.

0

1

2

3


```

|   Type = 14   |   Rsvd2   |   4 + n   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   EID-ML   |   RSVD3   |   EID-AFI = x   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     EID-Prefix ...   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   RLOC-AFI = x   |   Locator Address ...   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

EID-RLOC LCAF:

In which, for the purpose of NFV, EID prefix will be used to specify Function-Instance-EID, and Locator address is the RLOC associated with that Function-Instance-EID. This LCAF can be used in place of the Loc-AFI in the Map-Reply Message above to include a list of (Function-Instance-EID,RLOC) for every (Client-EID, Function-EID) in the Map-Reply.

Finally to store the affinity of the flow in the Mapping System a Map-Register can be used where EID AFI is filled with a LCAF type 12 (2-Tuple Src/Dst LCAF), and Loc-AFI is filled with the AFI of the Function-Instance-EID, and the Locator is filled with the Function-Instance-EID. This way, a query on the flow 2-Tuple returns the Function-Instance-EID that the flow is assigned to.

7. QOS and Echo Measurements

8. Security Considerations

there are no security considerations related with this memo.

9. IANA Considerations

there are no IANA considerations related with this memo.

10. Acknowledgements

11. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

Authors' Addresses

Sharon Barkai
ConteXtream Inc.
California
USA

Email: sbarkai@gmail.com

Dino Farinacci
California
USA

Email: farinacci@gmail.com

David Meyer
California
USA

Email: dmm@1-4-5.net

Fabio Maino
Cisco Systems
California
USA

Email: fmaino@cisco.com

Vina Ermagan
Cisco Systems
California
USA

Email: vermagan@cisco.com

LISP Working Group
Internet-Draft
Intended status: Experimental
Expires: July 13, 2019

S. Barkai
Fermi Serverless
D. Farinacci
lispers.net
D. Meyer
1-4-5.net
F. Maino
Cisco Systems
V. Ermagan
Google
A. Rodriguez-Natal
Cisco Systems
A. Cabellos-Aparicio
Technical University of Catalonia
January 9, 2019

LISP Based FlowMapping for Scaling NFV
draft-barkai-lisp-nfv-13

Abstract

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV). Network functions such as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware appliances embedded into the network as turn-key service-nodes or service-blades within routers. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled virtualized components of capacity and functionality. LISP-SDN based flow-mapping, dynamically assembles these components to whole solutions by steering the right traffic in the right sequence to the right virtual function instance.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute

working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 13, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Connectivity Model	5
4. Flow-Mapping Elements	7
5. Day-in-life of a Mapped Flow	8
5.1. XTR Flow Edge	9
5.2. Map Resolvers-Servers	11
5.3. XTRs-Mappers Scaling	11
6. Message Formats	11
7. QOS and Echo Measurements	14
8. Security Considerations	14
9. IANA Considerations	14
10. Acknowledgements	14
11. Normative References	14
Authors' Addresses	15

1. Introduction

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV).[RFC6830]Network functions such

as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware appliances embedded into the network as turn-key service-nodes or service-blades within routers.

This monolithic service delivery method increases the complexity of service roll-out and capacity planning, limits providers' choices, and slows down revenue generating service innovation. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled ("googlized") virtualized components of capacity and functionality. Such a component based model opens up service provider networks to the savings of elasticity and open architecture driven innovation. However this model also presents the network with the new challenges of assembling components, developed by 3rd parties, into whole solutions, by forwarding the right traffic to the right function-block at the right sequence.

While this is possible, to some extent, by traditional virtual networking - virtual bridges(vBridges) and virtual-routing-forwarding (VRF) - these mechanisms are relatively static and require complex and intensive configuration of network interfaces, while elastic components are not network topology bound. Software-defined-networks, (SDN) flow based models are much more dynamically programmable but are also very centralized and hence have limited scale and resiliency. By enhancing SDN models with RFC6830 overlay model we offer a best fit to dynamic assembly of virtualized network functions in the service-providers data-centers and distribution-centers.

2. Terminology

The following terms are used to describe a LISP based implementation of Software-Defined Flow-Mapping-Fabric for NFV:

- o LISP-SDN - is an enhancement to the basic SDN model of (1) hop-to-hop (2) push-down flow-commands (3) by concentrated-controller.. to a LISP based architecture of (1) distributed-overlay e.g. SDN over IP (2) based on a pull-publish-subscribe actions from xTR-edges up.. (3) to a global mapping service. A mapping service scaled by and connected over the IP underlay network. LISP-SDN lookup operation details are covered in [I-D.rodriqueznatal-lisp-multi-tuple-eids].
- o Virtualized Network Function (VNF) - is a process instance with an EID and RLOC that performs a defined set of inline network functions. a VNF can be software on a virtual-machine (VM) performing a function like multimedia signaling, mobility

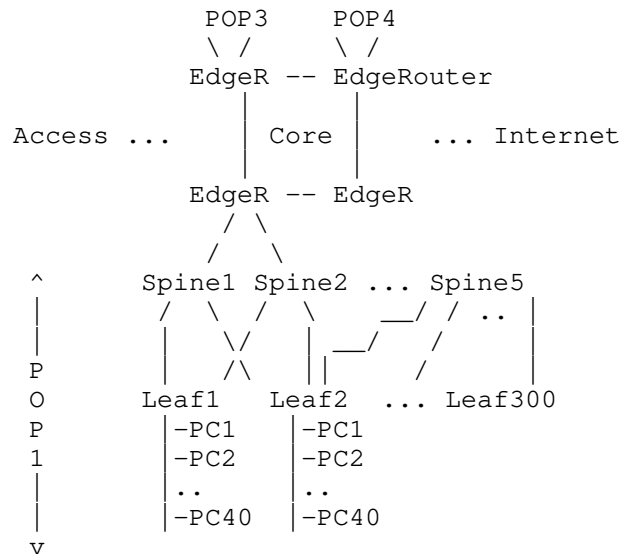
management, content caching or streaming, security, filtering, optimization, etc. A VNF class type and VNF instance capacity, load, and location are attributes that can be resolved by the LISP-SDN mapping service.

- o Client-Flow - is a sequence of packets that corresponds to a specific communication thread or network conversation between a client application and a network service. Client-flows are typically processed by various in-network functions either as the end service side to the network conversation, or as middle-box functionality.
- o SDN-xTR - is a LISP xTR that supports the lookup defined in [I-D.rodriqueznatal-lisp-multi-tuple-eids]. It classifies traffic into application flows, maps, encapsulates, and decapsulates flows in order to emerge a flow-mapping solution - along with a collection of the SDN-xTR elements, and the LISP-SDN mapping service.
- o SDN-Overlay - is the network formed by the collection of inter-connected SDN-xTR
- o SDN-Underlay - is the IPvN network connecting SDN-xTRs
- o SDN-Outerlay (interim name)- is the collection of networks and interfaces aggregated by the various SDN-xTRs connecting VNFs and Client-flows coming from access networks or the Internet.
- o Flow-Rule - is a set of pattern tuples that match any part of a packet header and is used to classify packets into flows as well as trigger forwarding actions such as encapsulation / decapsulation, network address translation (NAT), etc. We differentiate between exact-match rules (many) which include an exact set of tuple bits, and best match rules (fewer) which contain both tuple bits and wild-cards "*".
- o Virtual IP (VIP) - is an IP address or EID that identifies a function rather than a specific destination. For example all the encapsulated client-flow traffic sent from a base-station eNodeBs over a transport network, can have as destination a VIP which represents in a given LISP-SDN solution, the function mobile-gateway or PGW, and not any specific destination.
- o Flow-Affinity - is the association between a client-flow and a VNF instance. VNF logic will typically create long-lived (minutes) in memory states in order to perform its functions. Therefore once an affinity is established it is best to keep it for as long as possible in order not to stress or break the VNF application.

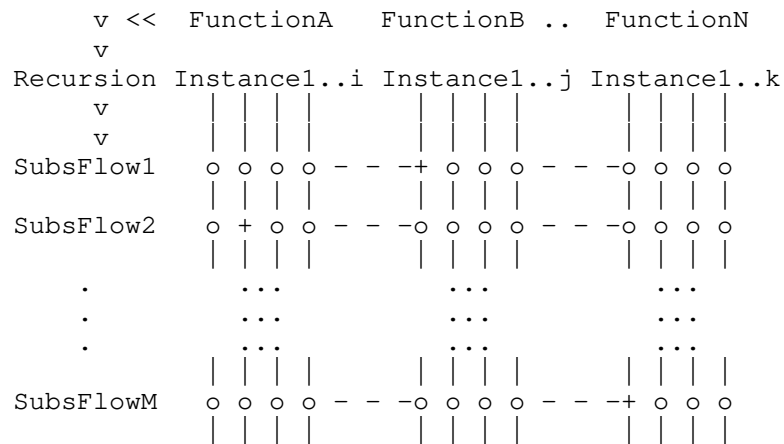
3. Connectivity Model

The basic connectivity model used to assemble VNFs into whole solution is the flow-mapping-fabric. Unlike topological forwarding which is based on source-subnet >> routed hop by hop >> destination-subnet, a flow-mapping-fabric maps, forwards and "patches" flows by identity directly to the end systems. The identities used for the flow-mapping-fabric are those associated with the client-flows e.g. Subscriber ID, phone number, TCP port, etc. and those associated with the VNF e.g. the type, location, physical address, etc. the flow-mapping-fabric is implemented as a LISP-SDN overlay, over in-place IP underlay, assembling outerlay flows into solutions. Bellow are basic assumptions regarding the Underlay, Outerlay, and Overlay in the solution:

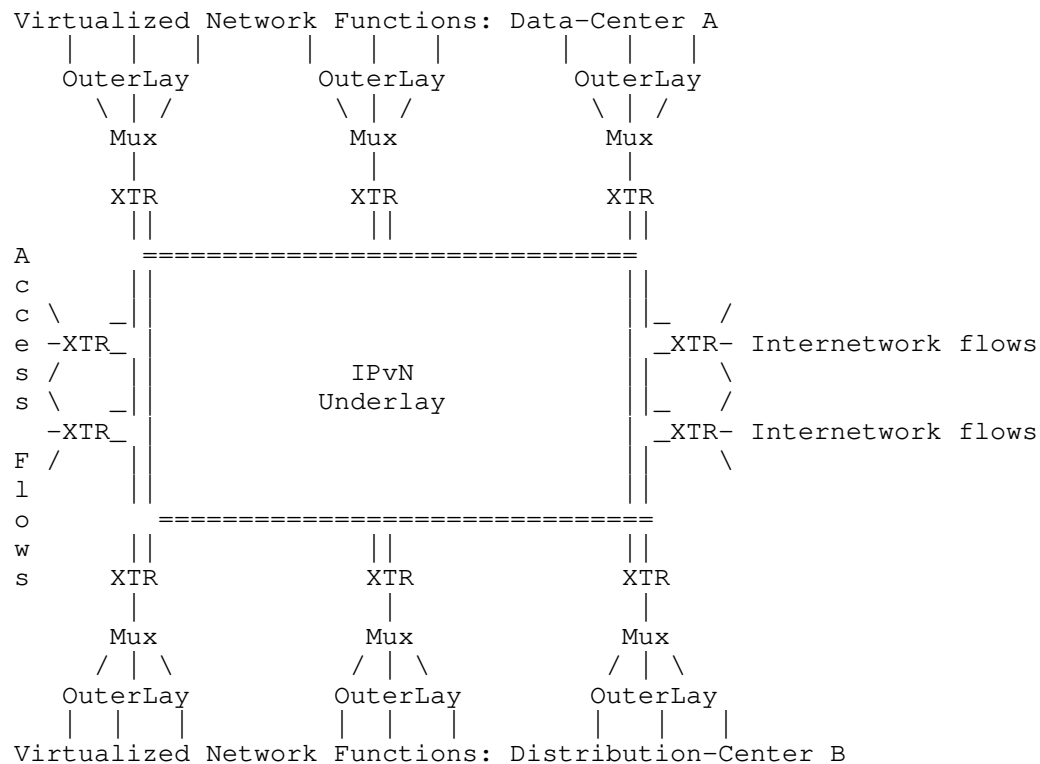
- o The underlying physical network is assumed to be topology based and implemented using standard bridging and routing. Conventional design principles are applied in order to achieve both capacity and availability of connectivity. Typical examples of underlays include spine-leaf switching for clustering server racks, and, core-edge routing inter-connecting server clusters across points of presence. Edge networks are also used to connect to access networks and Internet.
- o The flow-mapping-fabric maps outerlay client-flows to VNFs. This enables assembly, scaling, balanced high-utilization, massive concurrency, and hence, performance of NFVs. By mapping each client-flow to the correct functional instance the system engages as many VNF components as are available, scaled within and across data-centers. Applied recursively client-flow mapping can chain a sequence of VNF components to make up an end-to-end service.
- o The overlay network is based on location-identity-separation and forms a virtualization indirection ring around spines and cores. The overlay edges aggregate outerlay client-flows and VNFs. Outerlay flows are classified, mapped, and encapsulated over the edge through the underlay interfaces and are transported to the right identity's locations.



Core-Edge Spine-Leaf Underlays



Flow-Mapping-Fabric



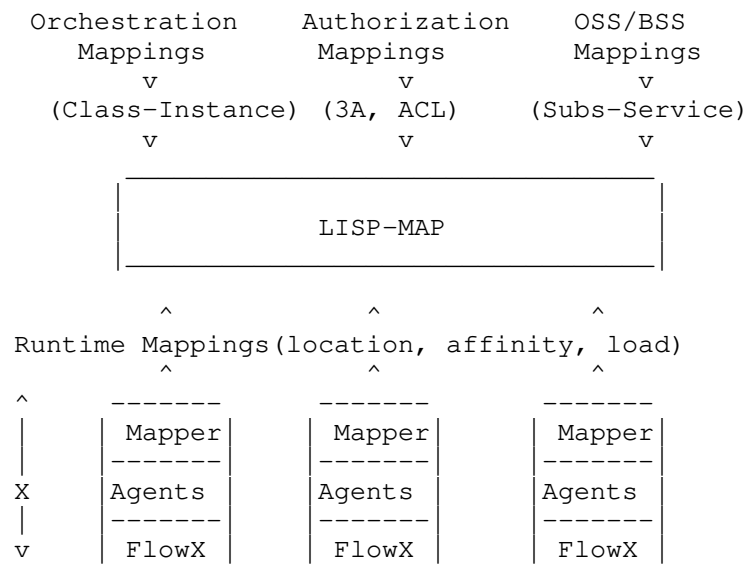
NFV Outerlay, LISP-SDN Overlay, IP Underlay

4. Flow-Mapping Elements

In order to implement NFV Flow-Mapping-Fabric using LISP-SDN We use the following components and capabilities:

1. **Flow-Switching:** is a component within an SDN-xTR and contains a set of n-tuple flow-rules matched against each packet in order to separate it to (LOCALLY defined) sequences representing flows. Flows are either Encapsulated into the Overlay, decapsulated to the Outerlay, or forwarded to SDN-xTR Control Agents.
2. **Control-Agents:** are software processes running in SDN-xTRs and are invoked for each flow where an exact match was not present in the Flow-Switching. The default "catch-all" Flow-Handler maps IP flows to locations and gateways based on RFC 6830. Protocol and application specific handlers can be loaded into the SDN-xTR for handling specific mapping and AFFINITY requirements of network functions. Examples of such protocols and applications can be SIP, GTP, S1X etc.

3. **Global-Mapping:** is how GLOBALLY significant key-value mappings is translated to LOCALLY defines flow masks and encapsulation actions. Examples of such mappings include: Map a functional instance ID to a function class ID; map subscriber-application ID to virtual function instance ID; map instance ID to location; instance to health, load, tenant; etc.



5. Day-in-life of a Mapped Flow

Let us walk through detailed steps of the use of RFC6830 and LISP architecture in order to perform resource virtualization and flow assignment to virtual function instances.

At a high level, when a client-flow packet first arrives at a SDN-xTR on the edge of the LISP overlay, the SDN-xTR must decide on a VNF instance that is best suited to service this flow, assign this flow to the selected VNF, and encapsulate this flow to the RLOC of the selected virtual function instance.

To select the best suited VNF instance, the SDN-xTR queries the Mapping System with the extracted identity parameters, both the client and the function EIDs, and receives the list of all VNF instances that represent that Function along with their RLOC and health-load attributes. The SDN-xTR runs local algorithms on the returned set to select the best suited virtual function instance.

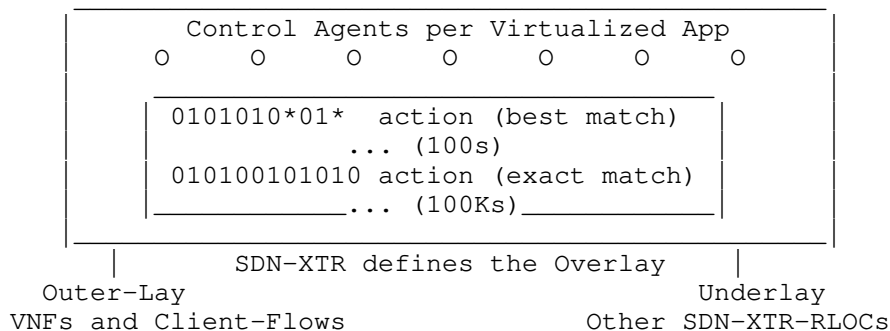
Once selected, the SDN-xTR stores (registers) the assignment of this flow to the associated VNF instance in the Mapping System. This assignment is referred to as the Affinity for this flow. The SDN-xTR also programs an exact match flow rule in its data-plane, so future packets from this flow will be mapped to the same EID-RLOC.

In the following subsections We describe this process in more detail.

5.1. XTR Flow Edge

SDN-xTR locations define the boundary of the virtual network. For the purpose of LISP-SDN flow-mapping-fabric We refer to the bellow SDN-XTR generic reference architecture. Actual vendor implementations may vary, but most likely will include similar components and structure. The SDN-XTR includes:

- o Mux-DeMux: Interfaces to the Underlay and Outerlay
- o Flow-Rules: Patterns-Actions, Exact / Best Match, Encap-Decap
- o Control-Agents: Application specific flow-handlers registered in the Flow-Rules



SDN-XTR Reference Architecture

SDN-XTR Flow Switching works as follows:

1. For traffic from the Outerlay of THIS xTR that has an exact match of all the source-dest-tags.. n-tuples, the packets are processed by rule actions including encapsulation to the RLOC of the xTR which aggregates the relevant function instance to which this flow is mapped to.

2. For traffic from the Underlay that has an exact match of all the source-dest-tags.. n-tuples, the packets are processed by rule actions including decapsulation and forwarding to the Outerlay of THIS xTR.
3. Traffic from the Outer-Lay or Underlay that does NOT have an exact match of all the source-dest-tags.. tuples required for normal forwarding, packets are forwarded to the control agent registered in the best-matching rule.

SDN-XTR Control Agents work as follows:

1. Mapping agent type and application scope is defined by the best match entries that point to it. Control agents will typically self-register in the flow-switch. XTR control-agents can register to an existing best-match rule, or instantiate a new one.
2. Typical rule-patterns are pattern-scoped by an agent registration, and can include: protocol or service type header indications; specific virtual IP addresses (VIP) that represent a service and not a specific destination; a specific source and wild-card destination; or vice versa.
3. Mapping agents work with the LISP-SDN mapping service in order to establish a global context and local considerations for mapping decision. The goal of the agents' decision is ultimately to provision the correct exact-match rule and actions that will offload the flow-packets to flow-switching described above.

The SDN-xTR control agents query the LISP-SDN Mapping System with the flow attributes including the destination VIP, as follows:

Mapping System Lookup: Map-Request (Client identity, Function-EID)

Two outcomes are possible based on whether an affinity already exists for this flow (flow has already been assigned to a virtual function instance):

- o Outcome A:
 - * If an affinity already exists in the Mapping System, the Mapping System returns the locator address (RLOC) associated with the Function-Instance-EID that the (Client-EID, Function-EID) is mapped to.
 - * Map-Reply: ((Client-EID, Function-EID) -> Function-Instance-RLOC)

- * In this case the Mapping System also subscribes the SDN-xTR to the Function-Instance-EID, and to the (Client-EID, Function-EID) flow in order to receive updates in case of changes on these entries. Examples of these changes are change of RLOC for the Function-Instance-EID (specially if this is a virtual application), or change of affinity for (Client-EID, Function-EID) to another Function-Instance-EID.
- * After receiving the Map-Reply from the Mapping System, the SDN-xTR programs an exact match for the flow in the xTR data-plane.
- o Outcome B:
 - * If there is no affinity previously stored, the Mapping System returns a list of Records, including one Record per each instance of the Function-EID, with their associated RLOCs and flags (weight, priority).
 - * Map-Reply: (client EID, Function-Instance-Record 1, Function-Instance-Record 2...)
 - * the SDN-xTR then selects the best suited Function-Instance-EID for this flow based on local algorithms, and registers the affinity in the Mapping System. The Mapping System stores the affinity and subscribes the SDN-xTR to the affinity and to the Function-Instance-EID in the affinity, so that SDN-xTR would receive updates if any of these changes.
 - * Map-Register ((Client-EID, Function-EID) -> Function-Instance-EID)
- o Note: An SDN-xTR must be able to query for the list of App-Instance-Records even if an affinity already exists. For this purpose a flag is required in the Map-Request to indicate whether xTR wants this info or not. We can overload the M bit in Map-Request, or allocate a new bit for this.

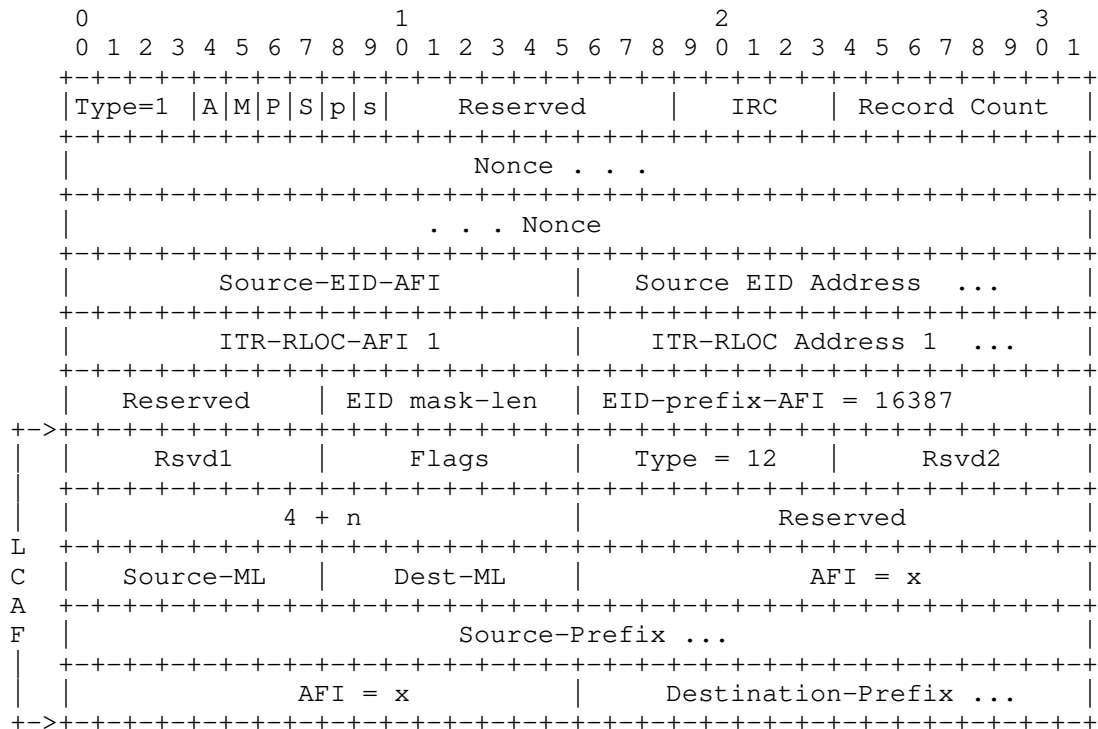
5.2. Map Resolvers-Servers

5.3. XTRs-Mappers Scaling

6. Message Formats

This section specifies the packet formats used throughout the flow-mapping process explained above. The lookup is based on what is described in [I-D.rodriqueznatal-lisp-multi-tuple-eids].

A Map-Request is used with a 2-Tuple Src/Dst LCAF to query the Mapping System for the affinity or list of virtual function instance records for this flow.



Where:

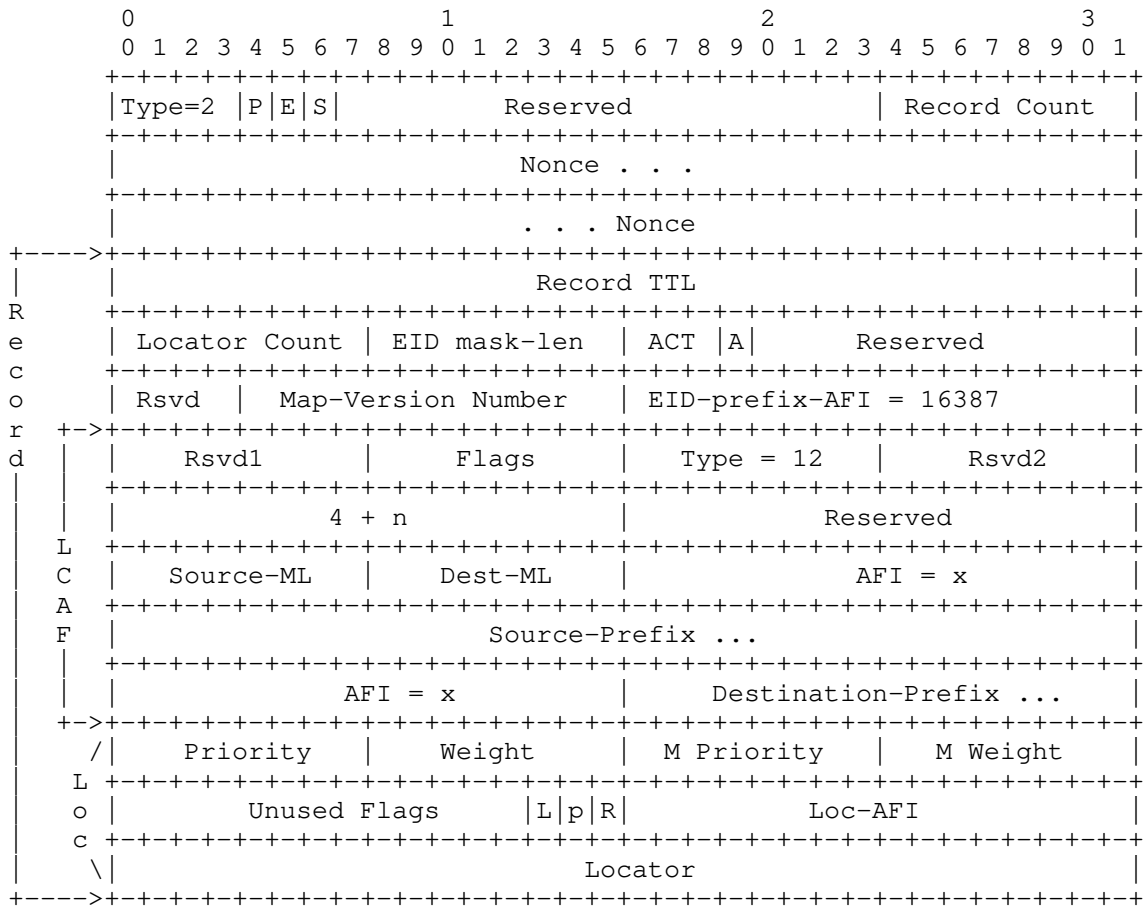
Source-Prefix = Client-EID

Destination-Prefix = App-EID

LISP Map-Request with 2-Tuple Src/Dst LCAF

In order to specify a 5 tuple flow, rather than just a two tuple source and destination, the combination of LCAF type 12 and LCAF type 4 must be used.

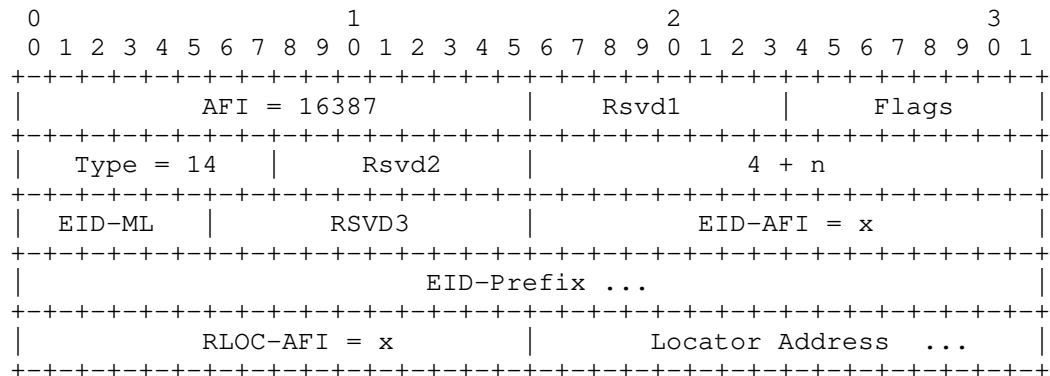
If an affinity exists in the Mapping System, meaning that the flow is already assigned to a virtual function instance, then the RLOC of that Function-Instance must be returned by the Mapping System. A Map-Reply with a 2-Tuple Src/Dst Lcaf can be used for this.



Map-Reply with 2-Tuple LCAF and Associated Function-Instance-RLOC

If no affinity exists, the Mapping System returns a list of records, including one record per each Function-Instance for the flow's Function-EID. A LISP Map-Reply can be used for this purpose with a 2-Tuple Src/Dst LCAF as the EID prefix in each Record.

If it is desired to return tuples of (Function-Instance-EID -> RLOC) per each record, a new LCAF, introduced as below, could be used.



EID-RLOC LCAF

In which, for the purpose of NFV, EID prefix will be used to specify Function-Instance-EID, and Locator address is the RLOC associated with that Function-Instance-EID. This LCAF can be used in place of the Loc-AFI in the Map-Reply Message above to include a list of (Function-Instance-EID,RLOC) for every (Client-EID, Function-EID) in the Map-Reply.

Finally to store the affinity of the flow in the Mapping System a Map-Register can be used where EID AFI is filled with a LCAF type 12 (2-Tuple Src/Dst LCAF), and Loc-AFI is filled with the AFI of the Function-Instance-EID, and the Locator is filled with the Function-Instance-EID. This way, a query on the flow 2-Tuple returns the Function-Instance-EID that the flow is assigned to.

7. QOS and Echo Measurements

8. Security Considerations

there are no security considerations related with this memo.

9. IANA Considerations

there are no IANA considerations related with this memo.

10. Acknowledgements

11. Normative References

- [I-D.rodriqueznatal-lisp-multi-tuple-eids]
Rodriguez-Natal, A., Cabellos-Aparicio, A., Barkai, S.,
Ermagan, V., Lewis, D., Maino, F., and D. Farinacci, "LISP
support for Multi-Tuple EIDs", draft-rodriqueznatal-lisp-
multi-tuple-eids-06 (work in progress), October 2018.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The
Locator/ID Separation Protocol (LISP)", RFC 6830,
DOI 10.17487/RFC6830, January 2013,
<<https://www.rfc-editor.org/info/rfc6830>>.

Authors' Addresses

Sharon Barkai
Fermi Serverless
CA
USA

Email: sharon@fermicloud.io

Dino Farinacci
lisppers.net
California
USA

Email: farinacci@gmail.com

David Meyer
1-4-5.net
USA

Email: dmm@1-4-5.net

Fabio Maino
Cisco Systems
California
USA

Email: fmaino@cisco.com

Vina Ermagan
Google
California
USA

Email: ermagan@gmail.com

Alberto Rodriguez-Natal
Cisco Systems
California
USA

Email: natal@cisco.com

Albert Cabellos-Aparicio
Technical University of Catalonia
Barcelona
Spain

Email: acabello@ac.upc.edu

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: January 16, 2014

F. Brockners
S. Bhandari
F. Maino
D. Lewis
Cisco
July 15, 2013

LISP Extensions for Segment Routing
draft-brockners-lisp-sr-00

Abstract

Segment Routing (SR) combines source routing and tunneling to steer traffic through the transit network. The Locator/ID Separation Protocol (LISP) separates IP addresses into Endpoint Identifiers (EIDs) and Routing Locators (RLOCs) and also leverages tunneling mechanisms. Mapping between EIDs and RLOCs is facilitated by the LISP mapping system. Combining LISP and SR enables the LISP mapping system to provide SR information to encapsulating routers so that traffic can be steered in the transit network or the list of segments a particular packet traverses is recorded in the packet header.

This document describes extensions required to the Locator/ID Separation Protocol (LISP) to enable a LISP mapping system to communicate list of segment identifiers or the request to record the list of segments a particular packet traverses to the encapsulating router.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 16, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Conventions	3
3. Use cases that combine LISP and SR	4
3.1. Traffic steering/traffic engineering	4
3.2. Traffic tracing	4
4. LISP extensions to support SR	5
4.1. Example ELPs	7
4.1.1. Example: ELP with only SR used	7
4.1.2. Example: ELP with SR and reencapsulating routers combined	7
5. IANA Considerations	8
6. Manageability Considerations	8
7. Security Considerations	9
8. Acknowledgements	9
9. References	9
9.1. Normative References	9
9.2. Informative References	9
Authors' Addresses	10

1. Introduction

Segment Routing (SR) allows for a flexible definition of end-to-end paths within network topologies by encoding paths as sequences of topological sub-paths, called "segments" as described in [I-D.filsfils-rtgwg-segment-routing]. Segment routing can be applied to IPv6 with a new type of routing extension header. The Locator/ID Separation Protocol [RFC6830] specifies an architecture and mechanism for replacing the addresses currently used by IP with two separate name spaces: Endpoint IDs (EIDs), used within sites; and Routing Locators (RLOCs), used on the transit networks that make up the Internet infrastructure. To achieve this separation, LISP defines protocol mechanisms for mapping from EIDs to RLOCs. In addition, LISP assumes the existence of a database to store and propagate those mappings globally.

When LISP is combined with SR, the EID to RLOC mapping information can be extended with segment routing information. This allows for a closer correlation between the transit network, that is sometimes also referred to as the underlay network, and the overlay network. It is beyond the scope of this document to describe how the LISP mapping system obtains a segment list for a particular EID-to-RLOC mapping. This draft outlines use-cases for combining LISP and SR as well as extensions to the LISP Canonical Address Format (LCAF) for traffic engineering (LCAF type 10) [I-D.ietf-lisp-lcaf]. These extensions are to be integrated into a future revision of [I-D.ietf-lisp-lcaf].

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document uses the Terminology as defined in [I-D.filsfils-rtgwg-segment-routing] and [I-D.ietf-lisp-lcaf].

Abbreviations used in this document:

AFI: Address Family Identifier

EID: Endpoint Identifier

ELP: Explicit Locator Path

ETR: Egress Tunnel Router

ITR: Ingress Tunnel Router

LCAF: LISP Canonical Address Format

LISP: Locator/ID Separation Protocol

OAM: Operation Administration Maintenance

RLOC: Routing Locator

SR: Segment Routing

SID: Segment Identifier

Segment List: Ordered list of segment identifiers

3. Use cases that combine LISP and SR

Use-cases that combine LISP and SR include traffic steering/traffic engineering as well as traffic tracing in the underlay network.

3.1. Traffic steering/traffic engineering

LISP combined with SR can be used to steer traffic in the underlay network: The mapping system communicates a segment list to the LISP ingress tunnel router (ITR) when resolving the EID-to-RLOC mapping as part of a LISP Map-Reply. This extension allows the LISP mapping system to provide a list of segment identifiers to encapsulating routers so that traffic can be steered in the transit network. In a typical setup the LISP ingress tunnel router would retrieve the segment list from the mapping system along with the associated RLOC using the EID as the lookup key. The ITR encapsulates the packet to the RLOC, also including the segment list in the segment routing extension header. The packet is forwarded to the ETR using segment routing techniques. The ETR decapsulates the packet and delivers the packet to the destination EID. Given that in SR with IPv6 transport the entire segment list is available in the SR-specific extension header of the outer IPv6 header, the LISP egress tunnel router, which is the tunnel endpoint is also informed about the path a particular packet took in the transport network.

LISP with SR for traffic engineering adds to the LISP traffic engineering use-cases described in [I-D.farinacci-lisp-te]. LISP combined with SR offers traffic engineering without using reencapsulating tunnels [RFC6830]. Reencapsulating tunnels and SR with LISP are complementary traffic engineering techniques and could be combined. SR could for example be used in an explicit locator path (ELP) to further traffic engineer a path between two reencapsulating routers.

3.2. Traffic tracing

LISP combined with SR can be used to get more information about the path a packet took in the underlay network without sending extra probe traffic. When SR is applied to IPv6, the segment list describing the path that a packet takes through the network can be recorded in the SR-specific extension header of the outer IPv6 packet header. This activity is referred to as segment tracing. Segment tracing can be performed independently from steering traffic using SR techniques. It can also be used in a transit network which performs normal IPv6 routing. When tracing is enabled, the segment ID of every segment that a packet traverses is recorded in the SR-specific extension header. This means that the egress tunnel router receives information about the path, represented by the segment list, a particular packet has taken in the underlay network. Different from OAM mechanisms which send active probe packets, tracing information can be made available for production traffic. The egress tunnel router can choose to provide the traced segment list back to the mapping system, for example through a LISP Map-Register. This information can be used to ensure path symmetry send/receive traffic in the transit network, or can serve other OAM or statistical purposes.

4. LISP extensions to support SR

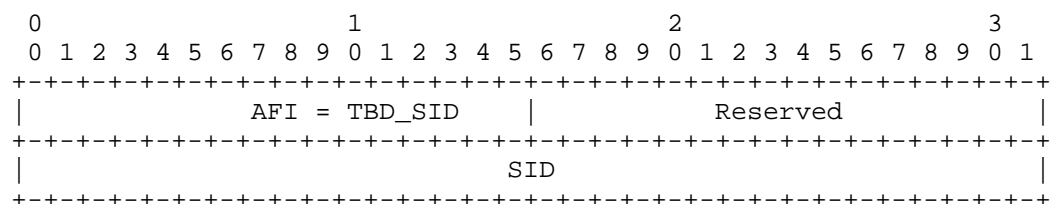
Segment routing information can be contained within the LISP mapping system. A segment identifier is a 32-bit identification either for a topological instruction or a service instruction. See [I-D.filsfils-rtgwg-segment-routing] for details.

An EID can be associated with one or multiple ordered lists of segment identifiers, also referred to as "segment lists", encoding the topological and service source route of a packet. The segment list can serve either traffic engineering or operational purposes. In case of traffic engineering purposes, the segment list describes the set of segments a packet visits when traversing the transit network. The segment list enables the ITR to steer traffic using segment routing techniques. The ITR retrieves the segment list from the mapping system along with the associated RLOC using the EID as the lookup key. For operations and maintenance use, the segment list documents the set of segments a packet visited on its way through the transit network. It is beyond the scope of this document to describe the detailed procedures how the LISP mapping system obtains a segment list for a particular EID-to-RLOC mapping.

Segment routing extensions for LISP extend the Explicit Locator Path (ELP) Canonical Address Format, which is LCAF type 10, see [I-D.ietf-lisp-lcaf] for details. A new Address Family Identifier (AFI) in LISP Canonical Address Format (LCAF) type [I-D.ietf-lisp-lcaf] carries the 32-bit segment identifier. For a

given EID lookup in the mapping database, the segment routing list in ELP LCAF type can be returned to provide a segment list to each locator in the Map-Reply locator set. The ELP LCAF type can also be used to send the segment list that a particular packet traversed to the LISP mapping system using a Map-Register message defined in [RFC6833].

The segment identification AFI to be allocated is described below:



AFI=TBD_SID: TBD_SID is a value allocated from [AFI] for segment identifiers.

Reserved: this 8-bit field reserved for future use and to carry specific control bits. If used with the ELP LCAF, this field carries several bits (see below).

SID: 4 byte segment identifier

The segment identification AFI is used within the ELP LCAF to describe the list of segments a packet is to visit or has visited on its way through the transit network. Further below examples are shown how the segment identification AFI is used for the ELP LCAF type. A new bit, the T-bit, is added to the LISP LCAF type 10 described in [I-D.ietf-lisp-lcaf]. This addition is to be integrated into a future revision of [I-D.ietf-lisp-lcaf].

For the segment routing AFI, the T-bit is defined as follows:

T-Bit: An additional bit in the Rsvd3 field is to be allocated in LCAF type 10. The T-bit (T=1) is used by the LISP mapping system to indicate to an ITR that for particular EID-to-RLOC mapping the segments traversed by packets SHOULD be recorded as a segment list in the SR IPv6 extension header. This bit is ignored if present in a Map-Register message. A Map-Register message could be used by the ETR to inform the mapping system about the segments that a packet visited in the transit network.

4.1. Example ELPs

4.1.1. Example: ELP with only SR used

This example shows the Explicit Locator Path (ELP) Canonical Address Format in a setup where segment routing is used in the transit network between ITR and ETR. Traffic engineering using reencapsulating routers is not used.

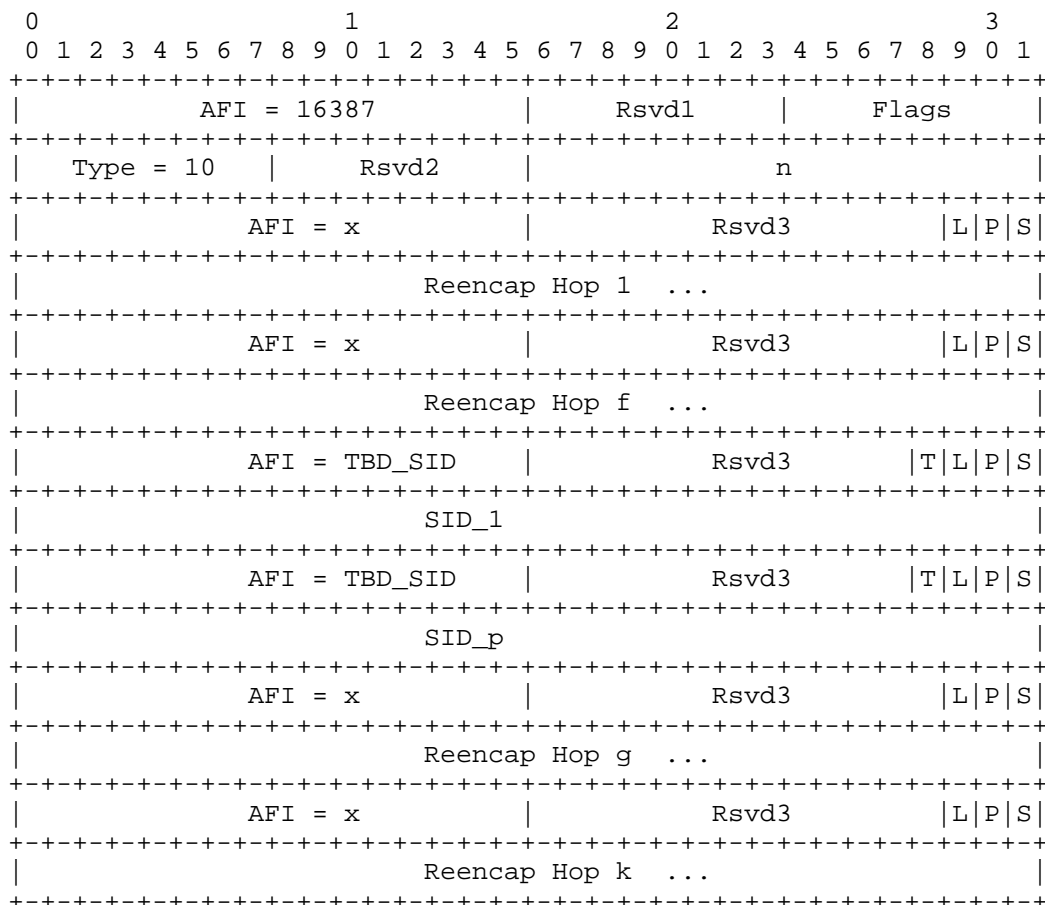
The reply to an EID-to-RLOC lookup contains the SIDs to be visited in the underlay network to reach the RLOC address returned in AFI=x. In the example below SID_1,...,SID_p are to be used for segment routing towards the "Address" RLOC. SID_p is the identifier of the last segment which takes the packet to the "Address" RLOC.

0										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								
AFI = 16387										Rsvd1										Flags																			
Type = 10					Rsvd2					n																													
AFI = TBD_SID										Rsvd3															T														
SID_1																																							
AFI = TBD_SID										Rsvd3															T														
SID_p																																							
AFI = x										Rsvd3																													
Address ...																																							

4.1.2. Example: ELP with SR and reencapsulating routers combined

This example shows the Explicit Locator Path (ELP) Canonical Address Format when using SR combined with reencapsulation routers.

Segment routing and traffic engineering using reencapsulating routers can be combined. In the example below, segment routing is used to steer traffic in the underlay between reencapsulating routers "f" and "g". There is no segment routing used between any of the other reencapsulating router hops.



5. IANA Considerations

A new AFI for segment identifiers is to be allocated by IANA (see [AFI] for a list of currently allocated AFIs).

6. Manageability Considerations

Manageability considerations will be addressed in a later version of this document..

7. Security Considerations

Security considerations will be addressed in a later version of this document.

8. Acknowledgements

The authors would like to thank Dino Farinacci for his input on this document.

9. References

9.1. Normative References

- [AFI] , "IANA, Address Family Identifier (AFIs), <http://www.iana.org/assignments/address-family-numbers/address-family-numbers.xhtml>", July 2013.
- [I-D.filsfils-rtgwg-segment-routing]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe, "Segment Routing Architecture", draft-filsfils-rtgwg-segment-routing-00 (work in progress), June 2013.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-02 (work in progress), March 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. Informative References

- [I-D.farinacci-lisp-te]
Farinacci, D., Lahiri, P., and M. Kowal, "LISP Traffic Engineering Use-Cases", draft-farinacci-lisp-te-03 (work in progress), July 2013.
- [I-D.filsfils-rtgwg-segment-routing-use-cases]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe, "Segment Routing Use Cases", draft-filsfils-rtgwg-segment-routing-use-cases-00 (work in progress), June 2013.
- [I-D.sivabalan-pce-segment-routing]

Sivabalan, S., Filsfils, C., Medved, J., Crabbe, E., and R. Raszuk, "PCE-Initiated Traffic Engineering Path Setup in Segment Routed Networks", draft-sivabalan-pce-segment-routing-00 (work in progress), June 2013.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

[RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.

Authors' Addresses

Frank Brockners
Cisco
Hansaallee 249, 3rd Floor
DUESSELDORF, NORDRHEIN-WESTFALEN 40549
Germany

Email: fbrockne@cisco.com

Shwetha Bhandari
Cisco
Cessna Business Park, Sarjapura Marathalli Outer Ring Road
Bangalore, KARNATAKA 560 087
India

Email: shwethab@cisco.com

Fabio Maino
Cisco
San Jose
USA

Email: fmaino@cisco.com

Darrel Lewis
Cisco
San Jose
USA

Email: darlewis@cisco.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: August 17, 2014

F. Brockners
S. Bhandari
F. Maino
D. Lewis
Cisco
February 13, 2014

LISP Extensions for Segment Routing
draft-brockners-lisp-sr-01

Abstract

Segment Routing (SR) combines source routing and tunneling to steer traffic through the transit network. The Locator/ID Separation Protocol (LISP) separates IP addresses into Endpoint Identifiers (EIDs) and Routing Locators (RLOCs) and also leverages tunneling mechanisms. Mapping between EIDs and RLOCs is facilitated by the LISP mapping system. Combining LISP and SR enables the LISP mapping system to provide SR information to encapsulating routers so that traffic can be steered in the transit network or the list of segments a particular packet traverses is recorded in the packet header.

This document describes extensions required to the Locator/ID Separation Protocol (LISP) to enable a LISP mapping system to communicate list of segment identifiers or the request to record the list of segments a particular packet traverses to the encapsulating router.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 17, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Conventions	3
3. Use cases that combine LIISP and SR	4
3.1. Traffic steering/traffic engineering	4
3.2. Traffic tracing	5
4. LIISP extensions to support SR	5
4.1. Deployment Scenario	7
4.2. Example ELPs	7
4.2.1. Example: ELP with only SR used	7
4.2.2. Example: ELP with SR and reencapsulating routers combined	8
5. IANA Considerations	9
6. Manageability Considerations	9
7. Security Considerations	9
8. Acknowledgements	10
9. Change log	10
10. References	10
10.1. Normative References	10
10.2. Informative References	10
Authors' Addresses	11

1. Introduction

Segment Routing (SR) allows for a flexible definition of end-to-end paths within network topologies by encoding paths as sequences of topological sub-paths, called "segments" as described in [I-D.filsfils-rtgwg-segment-routing]. Segment routing can be applied to IPv6 with a new type of routing extension header. The Locator/ID Separation Protocol [RFC6830] specifies an architecture and mechanism for replacing the addresses currently used by IP with two separate name spaces: Endpoint IDs (EIDs), used within sites; and Routing Locators (RLOCs), used on the transit networks that make up the Internet infrastructure. To achieve this separation, LISP defines protocol mechanisms for mapping from EIDs to RLOCs. In addition, LISP assumes the existence of a database to store and propagate those mappings globally.

When LISP is combined with SR, the EID to RLOC mapping information can be extended with segment routing information. This allows for a closer correlation between the transit network, that is sometimes also referred to as the underlay network, and the overlay network. It is beyond the scope of this document to describe how the LISP mapping system obtains a segment list for a particular EID-to-RLOC mapping. This draft outlines use-cases for combining LISP and SR as well as extensions to the LISP Canonical Address Format (LCAF) for traffic engineering (LCAF type 10) [I-D.ietf-lisp-lcaf]. These extensions are to be integrated into a future revision of [I-D.ietf-lisp-lcaf].

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document uses the Terminology as defined in [I-D.filsfils-rtgwg-segment-routing] and [I-D.ietf-lisp-lcaf].

Abbreviations used in this document:

AFI: Address Family Identifier

EID: Endpoint Identifier

ELP: Explicit Locator Path

ETR: Egress Tunnel Router

ITR: Ingress Tunnel Router

LCAF: LISP Canonical Address Format

LISP: Locator/ID Separation Protocol

OAM: Operation Administration Maintenance

RLOC: Routing Locator

SR: Segment Routing

SID: Segment Identifier

Segment List: Ordered list of segment identifiers

3. Use cases that combine LISP and SR

Use-cases that combine LISP and SR include traffic steering/traffic engineering as well as traffic tracing in the underlay network.

3.1. Traffic steering/traffic engineering

LISP combined with SR can be used to steer traffic in the underlay network: The mapping system communicates a segment list to the LISP ingress tunnel router (ITR) when resolving the EID-to-RLOC mapping as part of a LISP Map-Reply. This extension allows the LISP mapping system to provide a list of segment identifiers to encapsulating routers so that traffic can be steered in the transit network. In a typical setup the LISP ingress tunnel router would retrieve the segment list from the mapping system along with the associated RLOC using the EID as the lookup key. The ITR encapsulates the packet to the RLOC, also including the segment list in the segment routing extension header. The packet is forwarded to the ETR using segment routing techniques. The ETR decapsulates the packet and delivers the packet to the destination EID. Given that in SR with IPv6 transport the entire segment list is available in the SR-specific extension header of the outer IPv6 header, the LISP egress tunnel router, which is the tunnel endpoint is also informed about the path a particular packet took in the transport network.

LISP with SR for traffic engineering adds to the LISP traffic engineering use-cases described in [I-D.farinacci-lisp-te]. LISP combined with SR offers traffic engineering without using reencapsulating tunnels [RFC6830]. Reencapsulating tunnels and SR with LISP are complementary traffic engineering techniques and could be combined. SR could for example be used in an explicit locator

path (ELP) to further traffic engineer a path between two reencapsulating routers.

3.2. Traffic tracing

LISP combined with SR can be used to get more information about the path a packet took in the underlay network without sending extra probe traffic. When SR is applied to IPv6, the segment list describing the path that a packet takes through the network can be recorded in the SR-specific extension header of the outer IPv6 packet header. This activity is referred to as segment tracing. Segment tracing can be performed independently from steering traffic using SR techniques. It can also be used in a transit network which performs normal IPv6 routing. When tracing is enabled, the segment ID of every segment that a packet traverses is recorded in the SR-specific extension header. This means that the egress tunnel router receives information about the path, represented by the segment list, a particular packet has taken in the underlay network. Different from OAM mechanisms which send active probe packets, tracing information can be made available for production traffic. The egress tunnel router can choose to provide the traced segment list back to the mapping system, for example through a LISP Map-Register. This information can be used to ensure path symmetry send/receive traffic in the transit network, or can serve other OAM or statistical purposes.

4. LISP extensions to support SR

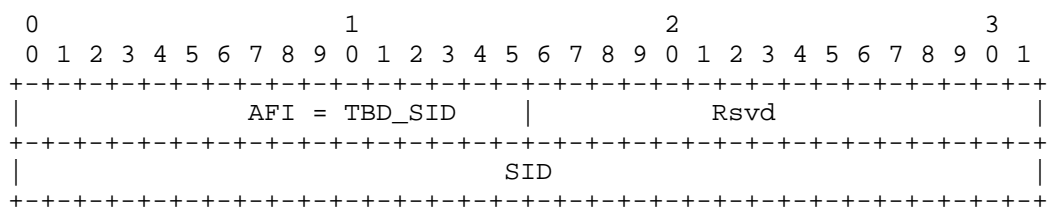
Segment routing information can be contained within the LISP mapping system. A segment identifier is a 32-bit identification either for a topological instruction or a service instruction. See [I-D.filsfils-rtgwg-segment-routing] for details.

An EID can be associated with one or multiple ordered lists of segment identifiers, also referred to as "segment lists", encoding the topological and service source route of a packet. The segment list can serve either traffic engineering or operational purposes. In case of traffic engineering purposes, the segment list describes the set of segments a packet visits when traversing the transit network. The segment list enables the ITR to steer traffic using segment routing techniques. For operations and maintenance use, the segment list documents the set of segments a packet visited on its way through the transit network. It is beyond the scope of this document to describe the detailed procedures how the LISP mapping system obtains a segment list for a particular EID-to-RLLOC mapping.

Segment routing extensions for LISP extend the Explicit Locator Path

(ELP) Canonical Address Format, which is LCAF type 10, see[I-D.ietf-lisp-lcaf] for details. A new Address Family Identifier (AFI) in LISP Canonical Address Format (LCAF) type [I-D.ietf-lisp-lcaf] is required to carry the 32-bit segment identifier. For a given EID lookup in the mapping database, the segment routing list in ELP LCAF type can be returned to provide a segment list to each locator in the Map-Reply locator set. The ELP LCAF type can also be used to send the segment list that a particular packet traversed to the LISP mapping system using a Map-Register message defined in [RFC6833].

The segment identification AFI to be allocated is described below:



AFI=TBD_SID: TBD_SID is a value allocated from [AFI] for segment identifiers.

Rsvd: should be set to zero and ignored.

SID: 4 byte segment identifier

The explicit path to be followed in the underlay is then encoded using the AFI for segment ID in the LISP LCAF type 10 described in [I-D.ietf-lisp-lcaf].

T-Bit: An additional bit in the Rsvd3 field is to be allocated in LCAF type 10. The T-bit (T=1) is used by the LISP mapping system to indicate to an ITR that for particular EID-to-RLOC mapping the segments traversed by packets SHOULD be recorded as a segment list in the SR IPv6 extension header. This bit is ignored if present in a Map-Register message. A Map-Register message could be used by the ETR to inform the mapping system about the segments that a packet visited in the transit network.

S-Bit: The S-bit SHOULD be set when AFI = TBD_SID.

P-Bit: The P-bit SHOULD be ignored when AFI = TBD_SID.

L-Bit: The L-bit SHOULD be ignored when AFI = TBD SID.

4.1. Deployment Scenario

As described in [RFC6833] LISP Mapping Service defines: the Map-Resolver, which accepts Map-Requests from an Ingress Tunnel Router (ITR) and "resolves" the EID-to-RLOC mapping using a mapping database; and the Map-Server, which learns authoritative EID-to-RLOC mappings from an Egress Tunnel Router (ETR) and publishes them in a database. The LISP Extensions for Segment Routing described in this document primarily apply to deployment scenarios where MAP-Server and MAP-Resolver have visibility into or interface with a system that has knowledge of the network topology and can determine paths from source to destination RLOCs. Implementations of the LISP mapping systems which complement Software Defined Networking (SDN) architectures, such as the implementation as part of the OpenDaylight project[ODLLISP] fall into this category. In these deployments the LISP mapping system can retrieve the necessary information related to topology and path selection to implement the extensions defined in this document. It allows the mapping system to provide the required information in the MAP resolve response to correlate overlay with underlay network and offer solutions to control the path taken in the underlay network.

4.2. Example ELPs

4.2.1. Example: ELP with only SR used

This example shows the Explicit Locator Path (ELP) Canonical Address Format in a setup where segment routing is used in the transit network between ITR and ETR. Traffic engineering using reencapsulating routers is not used.

The reply to an EID-to-RLOC lookup contains the SIDs to be visited in the underlay network to reach the RLOC address returned in AFI=x. In the example below SID_1,...,SID_p are to be used for segment routing towards the "Address" RLOC. SID_p is the identifier of the last segment which takes the packet to the "Address" RLOC.

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|               AFI = 16387               |   Rsvd1   |   Flags   |
+-----+-----+-----+-----+-----+-----+-----+-----+
|   Type = 10   |   Rsvd2   |               n               |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               AFI = TBD_SID               |   Rsvd3   | T | L | P | S |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               SID_1               |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               AFI = TBD_SID               |   Rsvd3   | T | L | P | S |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               SID_p               |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               AFI = x               |   Rsvd3   | T | L | P | S |
+-----+-----+-----+-----+-----+-----+-----+-----+
|               Address ...               |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

4.2.2. Example: ELP with SR and reencapsulating routers combined

This example shows the Explicit Locator Path (ELP) Canonical Address Format when using SR combined with reencapsulation routers.

Segment routing and traffic engineering using reencapsulating routers can be combined. In the example below, segment routing is used to steer traffic in the underlay between reencapsulating routers "f" and "g". There is no segment routing used between any of the other reencapsulating router hops.

0										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								
AFI = 16387										Rsvd1										Flags																			
Type = 10										Rsvd2										n																			
AFI = x										Rsvd3										T L P S																			
										Reencap Hop 1 ...																													
AFI = x										Rsvd3										T L P S																			
										Reencap Hop f ...																													
AFI = TBD_SID										Rsvd3										T L P S																			
SID_1																																							
AFI = TBD_SID										Rsvd3										T L P S																			
SID_p																																							
AFI = x										Rsvd3										T L P S																			
										Reencap Hop g ...																													
AFI = x										Rsvd3										T L P S																			
										Reencap Hop k ...																													

5. IANA Considerations

TBD.

6. Manageability Considerations

Manageability considerations will be addressed in a later version of this document..

7. Security Considerations

Security considerations will be addressed in a later version of this document.

8. Acknowledgements

The authors would like to thank Dino Farinacci, Erik Nordmark and participants of the LISP wg for their input on this document.

9. Change log

Changes from 00 - 01

- o Added a section on deployment scenario to clarify the applicability of the extension described in this draft.

10. References

10.1. Normative References

- [AFI] "IANA, Address Family Identifier (AFIs), <http://www.iana.org/assignments/address-family-numbers/address-family-numbers.xhtml>", July 2013.
- [I-D.filsfils-rtgwg-segment-routing]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe, "Segment Routing Architecture", draft-filsfils-rtgwg-segment-routing-00 (work in progress), June 2013.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-02 (work in progress), March 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

- [I-D.farinacci-lisp-te]
Farinacci, D., Lahiri, P., and M. Kowal, "LISP Traffic Engineering Use-Cases", draft-farinacci-lisp-te-03 (work in progress), July 2013.
- [I-D.filsfils-rtgwg-segment-routing-use-cases]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R.,

Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe,
"Segment Routing Use Cases",
draft-filsfils-rtgwg-segment-routing-use-cases-00 (work in
progress), June 2013.

[I-D.sivabalan-pce-segment-routing]
Sivabalan, S., Filsfils, C., Medved, J., Crabbe, E., and
R. Raszuk, "PCE-Initiated Traffic Engineering Path Setup
in Segment Routed Networks",
draft-sivabalan-pce-segment-routing-00 (work in progress),
June 2013.

[ODLLISP] "Open Day Light Lisp Flow Mapping, [https://
wiki.opendaylight.org/view/
OpenDaylight_Lisp_Flow_Mapping:Architecture](https://wiki.opendaylight.org/view/OpenDaylight_Lisp_Flow_Mapping:Architecture)", Feb 2014.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The
Locator/ID Separation Protocol (LISP)", RFC 6830,
January 2013.

[RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation
Protocol (LISP) Map-Server Interface", RFC 6833,
January 2013.

Authors' Addresses

Frank Brockners
Cisco
Hansaallee 249, 3rd Floor
DUESSELDORF, NORDRHEIN-WESTFALEN 40549
Germany

Email: fbrockne@cisco.com

Shwetha Bhandari
Cisco
Cessna Business Park, Sarjapura Marathalli Outer Ring Road
Bangalore, KARNATAKA 560 087
India

Email: shwethab@cisco.com

Fabio Maino
Cisco
San Jose
USA

Email: fmaino@cisco.com

Darrel Lewis
Cisco
San Jose
USA

Email: darlewis@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 24, 2014

L. Iannone
Telecom ParisTech
R. Jorgensen
Bredbandsfylket Troms
D. Conrad
Virtualized, LLC
October 21, 2013

LISP EID Block Management Guidelines
draft-iannone-lisp-eid-block-mgmt-03.txt

Abstract

This document proposes an allocation framework for the management of the LISP EID address prefix (requested in [I-D.ietf-lisp-eid-block]). The framework described relies on hierarchical distribution of the address space with sub-prefixes allocated on a temporary basis to requesting organizations.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 24, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Requirements Notation	3
2. Introduction	3
3. Definition of Terms	3
4. EID Prefix Allocation Policy	3
5. EID Prefixes Allocation Requirements	5
6. EID Prefix Request Template	6
7. General Considerations	6
8. Security Considerations	6
9. Acknowledgments	6
10. IANA Considerations	7
11. References	7
11.1. Normative References	7
11.2. Informative References	7
Appendix A. LISP Terms	8
Authors' Addresses	11

1. Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

The Locator/ID Separation Protocol (LISP - [RFC6830]) and related mechanisms ([RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC6837]) separates the IP addressing space into two logical spaces, the End-point Identifier (EID) space and the Routing Locator (RLOC) space. The first space is used to identify communication end-points, while the second is used to locate EIDs in the Internet routing infrastructure topology.

The document [I-D.ietf-lisp-eid-block] requested an IPv6 address block to be reserved for exclusive use for EID prefix allocation and assignment. The rationale, intent, size, and usage of the EID address block are described in [I-D.ietf-lisp-eid-block].

This document proposes an allocation framework for the EID address block based on temporary allocation of portions of the block to different requesting organizations.

3. Definition of Terms

The present document does not introduce any new term with respect to the set of LISP Specifications ([RFC6830], [RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC6837]). To help the reading of the present document the terminology introduced by LISP is summarized in Appendix A.

4. EID Prefix Allocation Policy

The allocation of EID prefixes MUST respect the following policies:

1. EID addressing prefixes are made available in the reserved space on a temporary basis and for experimental uses. The requester of an experimental prefix MUST provide a short description of the intended use or experiment that will be carried out (see Section 6). If the prefix will be used for activities not documented in the original description, the renewal of the allocation may be denied or withdrawn (see Section 5).

2. EID prefixes are allocated on a lease/license basis for a limited period of time (which can be renewed). The lease/license period SHOULD NOT be longer than one year.
3. Exception to the previous rule may be granted in cases in which the prefix has been delegated to an organization that will act as a registry for further sub-allocations. Sub-allocations MUST respect this present list of policies as well as the allocation requirements outlined in Section 5. Requests for a prefix delegation that will be used for further sub-allocations MUST clearly state such intent in the short description of the intended use document.
4. All of the allocations (renewed or not, including delegations and sub-allocations) MUST end by 31 December 2017, in accordance to the 3+3 years experimental allocation plan outlined in [I-D.ietf-lisp-eid-block].
5. Upon IETF review before 31 December 2017, the EID prefix space may become a permanent allocation. In this case existing allocations CAN be renewed and new allocations granted (still on a yearly temporary basis). All allocations (renewed or not, including delegations and sub-allocations) MUST end by 31 December 2020, in accordance to the 3+3 years plan outlined in [I-D.ietf-lisp-eid-block]. During the second 3 years phase of the experiment, the IETF will decide the final EID prefix block size and elaborate the allocation and management policies that will be applied starting 1 January 2021.
6. When an allocation is freed because of non-renewal or the termination of an experiment, the address space is returned to the global pool of free EID prefixes. This freed allocation MUST NOT be announced through registration on Map Servers in the LISP mapping system for at least 72 hours to ensure expiration of all cached map entries in the global LISP infrastructure.
7. The EID prefix of an allocation that is not renewed (or whose renewal has been denied) can be re-used after no less than one week from the date when the EID prefix is freed. This delay will provide sufficient time for all cached map entries in the global LISP infrastructure to expire and will allow any management process for re-allocation to be dealt with.
8. EID prefix allocations can be revoked as a result of abuse, unjustified usage (e.g., not conforming the intended use provided at request time), failure to pay maintenance fees, legal court orders, etc. Withdrawal can be enforced by filtering on Map Servers so to prevent map registration.

If/When the EID block experiment changes status (e.g., to not being "experimental"), and following the policies outlined in [RFC5226], the EID block will change status as well and will be converted to a permanent allocation. The IETF will define the transition process from the policies and requirements outlined in this document to a new set of policies and requirements. This transition process will include mechanisms that will allow for requests to convert existing temporary allocations (without renumbering) to permanent allocations.

5. EID Prefixes Allocation Requirements

All EID prefix allocations (and delegations) MUST respect the following requirements:

1. Allocations MUST be globally unique.
2. Requirements for allocation MUST be the same globally. No regional/national/local variations are permitted.
3. The minimum allocated prefix size MUST be a /48. An allocation may be larger (i.e., shorter prefix) provided that the requester is able to justify the intended size in their request description.
4. Registration information MUST be maintained and made publicly available through a searchable interface, preferably RDAP ([I-D.ietf-weirds-rdap-sec]) and optionally whois, http, or similar.
5. If fees are charged for EID allocation and registration services, those fees MUST be no more than the cost of providing those services.
6. Requesters obtaining an allocation SHOULD provide Reverse DNS service.
7. Requesters obtaining a delegation, hence acting as registries, MUST provide Reverse DNS service.
8. The service SHOULD be available 99% of the time.
9. Anyone, private persons, companies, or other entities can request EID space and those requests MUST be granted, provided that they can show a clear intent in carrying out LISP experimentation.

6. EID Prefix Request Template

Future versions of this document will include a detailed allocation (and delegation) request template to ensure a uniform process. An example of a similar template/process is the IANA Private Enterprise Number online request form (<http://pen.iana.org/pen/PenApplication.page>). The EID Prefix Request template MUST at minimum contain:

- o Requester Information (e.g., company name)
- o Requester Referral Person (and Contact Information)
- o Requested EID prefix size
- o Request Rationale

7. General Considerations

This document is a starting point for discussion aiming to address the concerns raised during the IETF Review of [I-D.ietf-lisp-eid-block], more specifically the lack of guidelines concerning the EID Block allocation and management.

Discussion with IANA, the RIR communities, and the IETF community should be carried out in order to verify compatibility of the proposed policy and agree upon the process for EID prefix allocation and management.

8. Security Considerations

This document does not introduce new security threats in the LISP architecture nor in the Legacy Internet architecture.

For accountability reasons, and in line with the security considerations in [RFC7020], each allocation request MUST contain accurate information on the requesting entity (company, institution, individual, etc.) and valid and accurate contact information of a referral person (see Section 6).

9. Acknowledgments

Thanks to J. Curran, A. Severin, B. Haberman, T. Manderson, D. Lewis, D. Farinacci, for their helpful comments.

10. IANA Considerations

This document provides only management guidelines for the reserved LISP EID prefix requested and allocated in [I-D.ietf-lisp-eid-block].

There is an operational requirement for an EID allocation service that ensures uniqueness of EIDs allocated according to the requirements described in Section 5. Furthermore, there is an operational requirement for EID registration service that allows a lookup of the contact information of the entity to which the EID was allocated.

IANA must ensure both of these services are provided, for the space directly allocated by IANA, in a globally uniform fashion for the duration of the experiment.

11. References

11.1. Normative References

- [I-D.ietf-lisp-eid-block]
Iannone, L., Lewis, D., Meyer, D., and V. Fuller, "LISP EID Block", draft-ietf-lisp-eid-block-05 (work in progress), August 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, August 2006.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

11.2. Informative References

- [I-D.ietf-weirds-rdap-sec]
Hollenbeck, S. and N. Kong, "Security Services for the Registration Data Access Protocol", draft-ietf-weirds-rdap-sec-05 (work in progress), August 2013.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, January 2013.
- [RFC7020] Housley, R., Curran, J., Huston, G., and D. Conrad, "The Internet Numbers Registry System", RFC 7020, August 2013.

Appendix A. LISP Terms

LISP operates on two name spaces and introduces several new network elements. This section provides high-level definitions of the LISP name spaces and network elements and as such, it must not be considered as an authoritative source. The reference to the authoritative document for each term is included in every term description.

Legacy Internet: The portion of the Internet that does not run LISP and does not participate in LISP+ALT or any other mapping system.

LISP site: A LISP site is a set of routers in an edge network that are under a single technical administration. LISP routers that reside in the edge network are the demarcation points to separate the edge network from the core network. See [RFC6830] for more details.

Endpoint ID (EID): An EID is a 32-bit (for IPv4) or 128-bit (for IPv6) value used in the source and destination address fields of the first (most inner) LISP header of a packet. A packet that is emitted by a system contains EIDs in its headers and LISP headers are prepended only when the packet reaches an Ingress Tunnel Router (ITR) on the data path to the destination EID. The source EID is obtained via existing mechanisms used to set a host's "local" IP address. An EID is allocated to a host from an EID-prefix block associated with the site where the host is located. See [RFC6830] for more details.

EID-prefix: A power-of-two block of EIDs that are allocated to a site by an address allocation authority. See [RFC6830] for more details.

EID-Prefix Aggregate: A set of EID-prefixes said to be aggregatable in the [RFC4632] sense. That is, an EID-Prefix aggregate is defined to be a single contiguous power-of-two EID-prefix block. A prefix and a length characterize such a block. See [RFC6830] for more details.

Routing LOCator (RLOC): A RLOC is an IPv4 or IPv6 address of an egress tunnel router (ETR). A RLOC is the output of an EID-to-RLOC mapping lookup. An EID maps to one or more RLOCs. Typically, RLOCs are numbered from topologically aggregatable blocks that are assigned to a site at each point to which it attaches to the global Internet; where the topology is defined by the connectivity of provider networks, RLOCs can be thought of as Provider Aggregatable (PA) addresses. See [RFC6830] for more details.

EID-to-RLOC Mapping: A binding between an EID-Prefix and the RLOC-set that can be used to reach the EID-Prefix. The general term "mapping" always refers to an EID-to-RLOC mapping. See [RFC6830] for more details.

Ingress Tunnel Router (ITR): An Ingress Tunnel Router (ITR) is a router that accepts receives IP packets from site end-systems on one side and sends LISP-encapsulated IP packets toward the Internet on the other side. The router treats the "inner" IP destination address as an EID and performs an EID-to-RLOC mapping lookup. The router then prepends an "outer" IP header with one of its globally routable RLOCs in the source address field and the result of the mapping lookup in the destination address field. See [RFC6830] for more details.

Egress Tunnel Router (ETR): An Egress Tunnel Router (ETR) receives LISP-encapsulated IP packets from the Internet on one side and sends decapsulated IP packets to site end-systems on the other side. An ETR router accepts an IP packet where the destination address in the "outer" IP header is one of its own RLOCs. The router strips the "outer" header and forwards the packet based on the next IP header found. See [RFC6830] for more details.

Proxy ITR (PITR): A Proxy-ITR (PITR) acts like an ITR but does so on behalf of non-LISP sites which send packets to destinations at LISP sites. See [RFC6832] for more details.

Proxy ETR (PETR): A Proxy-ETR (PETR) acts like an ETR but does so on behalf of LISP sites which send packets to destinations at non-LISP sites. See [RFC6832] for more details.

Map Server (MS): A network infrastructure component that learns EID-to-RLOC mapping entries from an authoritative source (typically an ETR). A Map Server publishes these mappings in the distributed mapping system. See [RFC6833] for more details.

Map Resolver (MR): A network infrastructure component that accepts LISP Encapsulated Map-Requests, typically from an ITR, quickly determines whether or not the destination IP address is part of the EID namespace; if it is not, a Negative Map-Reply is immediately returned. Otherwise, the Map Resolver finds the appropriate EID-to-RLOC mapping by consulting the distributed mapping database system. See [RFC6833] for more details.

The LISP Alternative Logical Topology (ALT): The virtual overlay network made up of tunnels between LISP+ALT Routers. The Border Gateway Protocol (BGP) runs between ALT Routers and is used to carry reachability information for EID-prefixes. The ALT provides a way to forward Map-Requests toward the ETR that "owns" an EID-prefix. See [RFC6836] for more details.

ALT Router: The device on which runs the ALT. The ALT is a static network built using tunnels between ALT Routers. These routers are deployed in a roughly-hierarchical mesh in which routers at each level in the topology are responsible for aggregating EID-Prefixes learned from those logically "below" them and advertising summary prefixes to those logically "above" them. Prefix learning and propagation between ALT Routers is done using BGP. When an ALT Router receives an ALT Datagram, it looks up the destination EID in its forwarding table (composed of EID-Prefix routes it learned from neighboring ALT Routers) and forwards it to the logical next-hop on the overlay network. The primary function of LISP+ALT routers is to provide a lightweight forwarding

infrastructure for LISP control-plane messages (Map-Request and Map-Reply), and to transport data packets when the packet has the same destination address in both the inner (encapsulating) destination and outer destination addresses ((i.e., a Data Probe packet). See [RFC6830] for more details.

Authors' Addresses

Luigi Iannone
Telecom ParisTech

Email: luigi.iannone@telecom-paristech.fr

Roger Jorgensen
Bredbandsfylket Troms

Email: rogerj@gmail.com

David Conrad
Virtualized, LLC

Email: drc@virtualized.org

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: August 29, 2016

L. Iannone
Telecom ParisTech
D. Lewis
Cisco Systems, Inc.
D. Meyer
Brocade
V. Fuller
February 26, 2016

LISP EID Block
draft-ietf-lisp-eid-block-13.txt

Abstract

This is a direction to IANA to allocate a /32 IPv6 prefix for use with the Locator/ID Separation Protocol (LISP). The prefix will be used for local intra-domain routing and global endpoint identification, by sites deploying LISP as EID (Endpoint Identifier) addressing space.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 29, 2016.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definition of Terms	3
3. Rationale and Intent	3
4. Expected use	4
5. Block Dimension	5
6. 3+3 Allocation Plan	6
7. Allocation Lifetime	7
8. Routing Considerations	7
9. Security Considerations	8
10. IANA Considerations	8
11. Acknowledgments	9
12. References	10
12.1. Normative References	10
12.2. Informative References	11
Appendix A. Document Change Log	12
Authors' Addresses	15

1. Introduction

This document directs the IANA to allocate a /32 IPv6 prefix for use with the Locator/ID Separation Protocol (LISP - [RFC6830]), LISP Map Server ([RFC6833]), LISP Alternative Topology (LISP+ALT - [RFC6836]) (or other) mapping systems, and LISP Interworking ([RFC6832]).

This block will be used as global Endpoint IDentifier (EID) space.

2. Definition of Terms

The present document does not introduce any new term with respect to the set of LISP Specifications ([RFC6830], [RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC6837]), but assumes that the reader is familiar with the LISP terminology. [I-D.ietf-lisp-introduction] provides an introduction to the LISP technology, including its terminology.

3. Rationale and Intent

Discussion within the LISP Working Group led to identify several scenarios in which the existence of a LISP specific address block brings technical benefits. Hereafter the most relevant scenarios are described:

Early LISP destination detection: With the current specifications, there is no direct way to detect whether or not a certain destination is in a LISP domain or not without performing a LISP mapping lookup. For instance, if an ITR is sending to all types of destinations (i.e., non-LISP destinations, LISP destinations not in the IPv6 EID block, and LISP destinations in the IPv6 EID block) the only way to understand whether or not to encapsulate the traffic is to perform a cache lookup and, in case of a LISP Cache miss, send a Map-Request to the mapping system. In the meanwhile (waiting the Map-Reply), packets may be dropped in order to avoid excessive buffering.

Avoid penalizing non-LISP traffic: In certain circumstances it might be desirable to configure a router using LISP features to natively forward all packets that have not a destination address in the block, hence, no lookup whatsoever is performed and packets destined to non-LISP sites are not penalized in any manner.

Traffic Engineering: In some deployment scenarios it might be desirable to apply different traffic engineering policies for LISP and non-LISP traffic. A LISP specific EID block would allow improved traffic engineering capabilities with respect to LISP vs. non-LISP traffic. In particular, LISP traffic might be identified without having to use DPI techniques in order to parse the encapsulated packet, performing instead a simple inspection of the outer header is sufficient.

Transition Mechanism: The existence of a LISP specific EID block may prove useful in transition scenarios. A non-LISP domain would ask for an allocation in the LISP EID block and use it to deploy LISP in its network. Such allocation will not be announced in the BGP routing infrastructure (cf., Section 4). This approach will allow non-LISP domains to avoid fragmenting their already allocated non-LISP addressing space, which may lead to BGP routing table inflation since it may (rightfully) be announced in the BGP routing infrastructure.

Limit the impact on BGP routing infrastructure: As described in the previous scenario, LISP adopters will avoid fragmenting their addressing space, since fragmentation would negatively impact the BGP routing infrastructure. Adopters will use addressing space from the EID block, which might be announced in large aggregates and in a tightly controlled manner only by proxy xTRs.

Is worth mentioning that new use cases can arise in the future, due to new and unforeseen scenarios.

Furthermore, the use of a dedicated address block will give a tighter control, especially filtering, over the traffic in the initial experimental phase, while facilitating its large-scale deployment.

[RFC3692] considers assigning experimental and testing numbers useful, and the request of a reserved IPv6 prefix is a perfect match of such practice. The present document follows the guidelines provided in [RFC3692], with one exception. [RFC3692] suggests the use of values similar to those called "Private Use" in [RFC5226], which by definition are not unique. One of the purposes of the present request to IANA is to guarantee uniqueness to the EID block. The lack thereof would result in a lack of real utility of a reserved IPv6 prefix.

4. Expected use

Sites planning to deploy LISP may request a prefix in the IPv6 EID

block. Such prefixes will be used for routing and endpoint identification inside the site requesting it. Mappings related to such prefix, or part of it, will be made available through the mapping system in use and registered to one or more Map Server(s).

The EID block must be used for LISP experimentation and must not be advertised in the form of more specific route advertisements in the non-LISP inter-domain routing environment. Interworking between the EID block sub-prefixes and the non-LISP Internet is done according to [RFC6832] and [RFC7215].

As the LISP adoption progresses, the EID block may potentially have a reduced impact on the BGP routing infrastructure, compared to the case of having the same number of adopters using global unicast space allocated by RIRs ([MobiArch2007]). From a short-term perspective, the EID block offers potentially large aggregation capabilities since it is announced by PxTRs possibly concentrating several contiguous prefixes. This trend should continue with even lower impact from a long-term perspective, since more aggressive aggregation can be used, potentially leading at using few PxTRs announcing the whole EID block ([FIABook2010]).

The EID block will be used only at configuration level, it is recommended not to hard-code in any way the IPv6 EID block in the router hardware. This allows avoiding locking out sites that may want to switch to LISP while keeping their own IPv6 prefix, which is not in the IPv6 EID block. Furthermore, in the case of a future permanent allocation, the allocated prefix may differ from the experimental temporary prefix allocated during the experimentation phase.

With the exception of Pitr case (described in Section 8) prefixes out of the EID block must not be announced in the BGP routing infrastructure.

5. Block Dimension

The working group reached consensus on an initial allocation of a /32 prefix. The reason of such consensus is manifold:

- o The working group agreed that /32 prefix is sufficiently large to cover initial allocation and requests for prefixes in the EID space in the next few years for very large-scale experimentation and deployment.
- o As a comparison, it is worth mentioning that the current LISP Beta Network ([BETA]) is using a /32 prefix, with more than 250 sites

using a /48 sub prefix. Hence, a /32 prefix appears sufficiently large to allow the current deployment to scale up and be open for interoperation with independent deployments using EIDs in the new /32 prefix.

- o A /32 prefix is sufficiently large to allow deployment of independent (commercial) LISP enabled networks by third parties, but may as well boost LISP experimentation and deployment.
- o The use of a /32 prefix is in line with previous similar prefix allocation for tunneling protocols ([RFC3056]).

6. 3+3 Allocation Plan

This document requests IANA to initially assign a /32 prefix out of the IPv6 addressing space for use as EID in LISP (Locator/ID Separation Protocol).

IANA allocates the requested address space by MMMM/YYYY0 for a duration of 3 (three) initial years (through MMMM/YYYY3), with an option to extend this period by 3 (three) more years (until MMMM/YYYY6). By the end of the first period, the IETF will provide a decision on whether to transform the prefix in a permanent assignment or to put it back in the free pool (see Section 7 for more information).

[RFC Editor: please replace MMMM and all its occurrences in the document with the month of publication as RFC.]

[RFC Editor: please replace YYYY0 and all its occurrences in the document with the year of publication as RFC.]

[RFC Editor: please replace YYYY3 and all its occurrences in the document with the year of publication as RFC plus 3 years, e.g., if published in 2016 then put 2019.]

[RFC Editor: please replace YYYY6 and all its occurrences in the document with the year of publication as RFC plus 6 years, e.g., if published in 2016 then put 2022.]

In the first case, i.e., if the IETF decides to transform the block in a permanent allocation, the EID block allocation period will be extended for three years (until MMMM/YYYY6) so to give time to the IETF to define the final size of the EID block and create a transition plan. The transition of the EID block into a permanent allocation has the potential to pose policy issues (as recognized in [RFC2860], section 4.3) and hence discussion with the IANA, the RIR

communities, and the IETF community will be necessary to determine appropriate policy for permanent EID block allocation and management. Note as well that the final permanent allocation may differ from the initial experimental assignment, hence, it is recommended not to hard-code in any way the experimental EID block on LISP-capable devices.

In the latter case, i.e., if the IETF decides to stop the EID block experimental use, by MMMM/YYYY3 all temporary prefix allocations in such address range must expire and be released, so that the entire /32 is returned to the free pool.

The allocation and management of the EID block for the initial 3 years period (and the optional 3 more years) is detailed in [I-D.ietf-lisp-eid-block-mgmt].

7. Allocation Lifetime

If no explicit action is carried out by the end of the experiment (by MMMM/YYYY3) it is automatically considered that there was no sufficient interest in having a permanent allocation and the address block will be returned to the free pool.

Otherwise, if the LISP Working Group recognizes that there is value in having a permanent allocation then explicit action is needed.

In order to trigger the process for a permanent allocation a document is required. Such document has to articulate the rationale why a permanent allocation would be beneficial. More specifically, the document has to detail the experience gained during experimentation and all of the technical benefits provided by the use of a LISP specific prefix. Such technical benefits are expected to lay in the scenarios described in Section 3, however, new unforeseen benefits may appear during experimentation. The description should be sufficiently articulate so to allow to provide an estimation of what should be the size of the permanent allocation. Note however that, as explained in Section 6, it is up to IANA to decide which address block will be used as permanent allocation and that such block may be different from the temporary experimental allocation.

8. Routing Considerations

In order to provide connectivity between the Legacy Internet and LISP sites, PITRs announcing large aggregates (ideally one single large aggregate) of the IPv6 EID block could be deployed. By doing so, PITRs will attract traffic destined to LISP sites in order to

encapsulate and forward it toward the specific destination LISP site. Routers in the Legacy Internet must treat announcements of prefixes from the IPv6 EID block as normal announcements, applying best current practice for traffic engineering and security.

Even in a LISP site, not all routers need to run LISP elements. In particular, routers that are not at the border of the local domain, used only for intra-domain routing, do not need to provide any specific LISP functionality but must be able to route traffic using addresses in the IPv6 EID block.

For the above-mentioned reasons, routers that do not run any LISP element, must not include any special handling code or hardware for addresses in the IPv6 EID block. In particular, it is recommended that the default router configuration does not handle such addresses in any special way. Doing differently could prevent communication between the Legacy Internet and LISP sites or even break local intra-domain connectivity.

9. Security Considerations

This document does not introduce new security threats in the LISP architecture nor in the legacy Internet architecture.

10. IANA Considerations

This document instructs the IANA to assign a /32 IPv6 prefix for use as the global LISP EID space using a hierarchical allocation as outlined in [RFC5226] and summarized in Table 1.

This document does not specify any specific value for the requested address block but suggests that should come from the 2000::/3 Global Unicast Space. IANA is not requested to issue an AS0 ROA (Route Origin Attestation [RFC6491]), since the Global EID Space will be used for routing purposes.

Attribute	Value
Address Block	2001:5::/32
Name	EID Space for LISP
RFC	[This Document]
Allocation Date	2015
Termination Date	MMMM/YYYY3 [1]
Source	True [2]
Destination	True
Forwardable	True
Global	True
Reserved-by-protocol	True [3]

[1] According to the 3+3 Plan outlined in this document termination date can be postponed to MMMM/YYYY6. [2] Can be used as a multicast source as well. [3] To be used as EID space by LISP [RFC6830] enabled routers.

Table 1: Global EID Space

[IANA: Please update the Termination Date and footnote [1] in the Special-Purpose Address Registry when the I-D is published as RFC.]

The reserved address space is requested for a period of time of three initial years starting in MMMM/YYYY0 (until MMMM/YYYY3), with an option to extend it by three years (until MMMM/YYYY6) up on decision of the IETF (see Section 6 and Section 7). Following the policies outlined in [RFC5226], upon IETF Review, by MMMM/YYYY3 decision should be made on whether to have a permanent EID block assignment. If no explicit action is taken or if the IETF review outcome will be that is not worth to have a reserved prefix as global EID space, the whole /32 will be taken out from the IPv6 Special Purpose Address Registry and put back in the free pool managed by IANA.

Allocation and management of the Global EID Space is detailed in a different document. Nevertheless, all prefix allocations out of this space must be temporary and no allocation must go beyond MMMM/YYYY3 unless the IETF Review decides for a permanent Global EID Space assignment.

11. Acknowledgments

Special thanks to Roque Gagliano for his suggestions and pointers. Thanks to Alvaro Retana, Deborah Brungard, Ron Bonica, Damien Saucez, David Conrad, Scott Bradner, John Curran, Paul Wilson, Geoff Huston,

Wes George, Arturo Servin, Sander Steffann, Brian Carpenter, Roger Jorgensen, Terry Manderson, Brian Haberman, Adrian Farrel, Job Snijders, Marla Azinger, Chris Morrow, and Peter Schoenmaker, for their insightful comments. Thanks as well to all participants to the fruitful discussions on the IETF mailing list.

The work of Luigi Iannone has been partially supported by the ANR-13-INFR-0009 LISP-Lab Project (www.lisp-lab.org) and the EIT KIC ICT-Labs SOFNETS Project.

12. References

12.1. Normative References

- [I-D.ietf-lisp-eid-block-mgmt] Iannone, L., Jorgensen, R., Conrad, D., and G. Huston, "LISP EID Block Management Guidelines", draft-ietf-lisp-eid-block-mgmt-06 (work in progress), August 2015.
- [RFC2860] Carpenter, B., Baker, F., and M. Roberts, "Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority", RFC 2860, DOI 10.17487/RFC2860, June 2000, <<http://www.rfc-editor.org/info/rfc2860>>.
- [RFC3692] Narten, T., "Assigning Experimental and Testing Numbers Considered Useful", BCP 82, RFC 3692, DOI 10.17487/RFC3692, January 2004, <<http://www.rfc-editor.org/info/rfc3692>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <<http://www.rfc-editor.org/info/rfc5226>>.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <<http://www.rfc-editor.org/info/rfc6830>>.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, DOI 10.17487/RFC6831, January 2013, <<http://www.rfc-editor.org/info/rfc6831>>.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller,

"Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, DOI 10.17487/RFC6832, January 2013, <<http://www.rfc-editor.org/info/rfc6832>>.

- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, DOI 10.17487/RFC6833, January 2013, <<http://www.rfc-editor.org/info/rfc6833>>.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <<http://www.rfc-editor.org/info/rfc6834>>.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, DOI 10.17487/RFC6835, January 2013, <<http://www.rfc-editor.org/info/rfc6835>>.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, DOI 10.17487/RFC6836, January 2013, <<http://www.rfc-editor.org/info/rfc6836>>.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, DOI 10.17487/RFC6837, January 2013, <<http://www.rfc-editor.org/info/rfc6837>>.

12.2. Informative References

- [BETA] LISP Beta Network, "<http://www.lisp4.net>".
- [FIABook2010] L. Iannone, T. Leva, "Modeling the economics of Loc/ID Separation for the Future Internet.", Towards the Future Internet - Emerging Trends from the European Research, Pages 11-20, ISBN: 9781607505389, IOS Press , May 2010.
- [I-D.ietf-lisp-introduction] Cabellos-Aparicio, A. and D. Saucez, "An Architectural Introduction to the Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-introduction-13 (work in progress), April 2015.
- [MobiArch2007] B. Quoitin, L. Iannone, C. de Launois, O. Bonaventure,

"Evaluating the Benefits of the Locator/Identifier Separation", The 2nd ACM-SIGCOMM International Workshop on Mobility in the Evolving Internet Architecture (MobiArch'07) , August 2007.

- [RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, DOI 10.17487/RFC3056, February 2001, <<http://www.rfc-editor.org/info/rfc3056>>.
- [RFC6491] Manderson, T., Vegoda, L., and S. Kent, "Resource Public Key Infrastructure (RPKI) Objects Issued by IANA", RFC 6491, DOI 10.17487/RFC6491, February 2012, <<http://www.rfc-editor.org/info/rfc6491>>.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", RFC 7215, DOI 10.17487/RFC7215, April 2014, <<http://www.rfc-editor.org/info/rfc7215>>.

Appendix A. Document Change Log

[RFC Editor: Please remove this section on publication as RFC]

Version 13 Posted MMMM 2016.

- o Changed I-D type from "Informational" to "Experimental" as requested by A. Retana during IESG review.
- o Dropped the appendix "LISP Terminology"; replaced by pointer to the LISP Introduction document.
- o Added Section 7 to clarify the process after the 3 years experimental allocation.
- o Modified the dates, introducing variables, so to allow RFC Editor to easily update dates by publication as RFC.

Version 12 Posted May 2015.

- o Fixed typos and references as suggested by the Gen-ART and OPS-DIR review.

Version 11 Posted April 2015.

- o In Section 4, deleted contradictory text on EID prefix advertisement in non-LISP inter-domain routing environments.

- o In Section 3 deleted the "Avoid excessive stretch" bullet, because confusing.
- o Deleted last bullet of the list in Section 3 because redundant w.r.t. global content of the document.

Version 10 Posted January 2015.

- o Keep alive version

Version 09 Posted July 2014.

- o Few Editorial modifications as requested by D. Saucez, as shepherd, during the write up of the document.
- o Allocation date postponed to beginning 2015, as suggested by D. Saucez.

Version 08 Posted January 2014.

- o Modified Section 4 as suggested by G. Houston.

Version 07 Posted November 2013.

- o Modified the document so to request a /32 allocation, as for the consensus reached during IETF 88th.

Version 06 Posted October 2013.

- o Clarified the rationale and intent of the EID block request with respect to [RFC3692], as suggested by S. Bradner and J. Curran.
- o Extended Section 3 by adding the transition scenario (as suggested by J. Curran) and the TE scenario. The other scenarios have been also edited.
- o Section 6 has been re-written to introduce the 3+3 allocation plan as suggested by B. Haberman and discussed during 86th IETF.
- o Section 10 has also been updated to the 3+3 years allocation plan.
- o Moved Section 11 at the end of the document.
- o Changed the original Definition of terms to an appendix.

Version 05 Posted September 2013.

- o No changes.

Version 04 Posted February 2013.

- o Added Table 1 as requested by IANA.
- o Transformed the prefix request in a temporary request as suggested by various comments during IETF Last Call.
- o Added discussion about short/long term impact on BGP in Section 4 as requested by B. Carpenter.

Version 03 Posted November 2012.

- o General review of Section 5 as requested by T. Manderson and B. Haberman.
- o Dropped RFC 2119 Notation, as requested by A. Farrel and B. Haberman.
- o Changed "IETF Consensus" to "IETF Review" as pointed out by Roque Gagliano.
- o Changed every occurrence of "Map-Server" and "Map-Resolver" with "Map Server" and "Map Resolver" to make the document consistent with [RFC6833]. Thanks to Job Snijders for pointing out the issue.

Version 02 Posted April 2012.

- o Fixed typos, nits, references.
- o Deleted reference to IANA allocation policies.

Version 01 Posted October 2011.

- o Added Section 5.

Version 00 Posted July 2011.

- o Updated section "IANA Considerations"
- o Added section "Rationale and Intent" explaining why the EID block allocation is useful.
- o Added section "Expected Use" explaining how sites can request and use a prefix in the IPv6 EID Block.

- o Added section "Action Plan" suggesting IANA to avoid allocating address space adjacent the allocated EID block in order to accommodate future EID space requests.
- o Added section "Routing Consideration" describing how routers not running LISP deal with the requested address block.
- o Added the present section to keep track of changes.
- o Rename of draft-meyer-lisp-eid-block-02.txt.

Authors' Addresses

Luigi Iannone
Telecom ParisTech

Email: ggx@gigix.net

Darrel Lewis
Cisco Systems, Inc.

Email: darlewis@cisco.com

David Meyer
Brocade

Email: dmm@1-4-5.net

Vince Fuller

Email: vaf@vaf.net

Network Working Group
Internet-Draft
Intended status: Informational
Expires: 24 March 2022

A. Cabellos
UPC-BarcelonaTech
D. Saucez (Ed.)
Inria
20 September 2021

An Architectural Introduction to the Locator/ID Separation Protocol
(LISP)
draft-ietf-lisp-introduction-15

Abstract

This document describes the architecture of the Locator/ID Separation Protocol (LISP), making it easier to read the rest of the LISP specifications and providing a basis for discussion about the details of the LISP protocols. This document is used for introductory purposes, more details can be found in [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis], the protocol specifications.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 24 March 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definition of Terms	4
3. LISP Architecture	5
3.1. Design Principles	5
3.2. Overview of the Architecture	6
3.3. Data-Plane	8
3.3.1. LISP Encapsulation	9
3.3.2. LISP Forwarding State	10
3.4. Control-Plane	10
3.4.1. LISP Mappings	10
3.4.2. Mapping System Interface	11
3.4.3. Mapping System	12
3.5. Internetworking Mechanisms	14
4. LISP Operational Mechanisms	15
4.1. Cache Management	15
4.2. RLOC Reachability	16
4.3. ETR Synchronization	17
4.4. MTU Handling	17
5. Mobility	18
6. Multicast	19
7. Use Cases	20
7.1. Traffic Engineering	20
7.2. LISP for IPv6 Co-existence	20
7.3. LISP for Virtual Private Networks	21
7.4. LISP for Virtual Machine Mobility in Data Centers	21
8. Security Considerations	21
9. IANA Considerations	23
10. Acknowledgements	23
11. References	23
11.1. Normative References	23
11.2. Informative References	26
Appendix A. A Brief History of Location/Identity Separation . .	27
A.1. Old LISP Models	28
Authors' Addresses	28

1. Introduction

This document introduces the Locator/ID Separation Protocol (LISP) architecture ([I-D.ietf-lisp-rfc6830bis], [I-D.ietf-lisp-rfc6833bis]), its main operational mechanisms and its design rationale. Fundamentally, LISP is built following a well-known architectural idea: decoupling the IP address overloaded semantics. Indeed and as pointed out by Noel Chiappa [RFC4984], currently IP addresses both identify the topological location of a network attachment point as well as the node's identity. However, nodes and routing have fundamentally different requirements. On the one hand, routing systems require that addresses are aggregatable and have topological meaning, on the other hand, nodes require to be identified independently of their current location [RFC4984].

LISP creates two separate namespaces, EIDs (End-host IDentifiers) and RLOCs (Routing LOCators), both are syntactically identical to the current IPv4 and IPv6 addresses. However, EIDs are used to uniquely identify nodes irrespective of their topological location and are typically routed intra-domain. RLOCs are assigned topologically to network attachment points and are typically routed inter-domain. With LISP, the edge of the Internet (where the nodes are connected) and the core (where inter-domain routing occurs) can be logically separated. LISP-capable routers interconnect the two logical spaces. LISP also introduces a database, called the Mapping System, to store and retrieve mappings between identity and location. LISP-capable routers exchange packets over the Internet core by encapsulating them to the appropriate location.

In summary:

- * RLOCs have meaning only in the underlay network, that is the underlying core routing system.
- * EIDs have meaning only in the overlay network, which is the encapsulation relationship between LISP-capable routers.
- * The LISP edge maps EIDs to RLOCs
- * Within the underlay network, RLOCs have both locator and identifier semantics
- * An EID within a LISP site carries both identifier and locator semantics to other nodes within that site
- * An EID within a LISP site carries identifier and limited locator semantics to nodes at other LISP sites (i.e., enough locator information to tell that the EID is external to the site)

The relationship described above is not unique to LISP and it is common to other overlay technologies.

The initial motivation in the LISP effort is to be found in the routing scalability problem [RFC4984], where, if LISP were to be completely deployed, the Internet core would be populated with RLOCs while Traffic Engineering mechanisms would be pushed to the Mapping System. In such scenario RLOCs are quasi-static (i.e., low churn), hence making the routing system scalable [Quoitin], while EIDs can roam anywhere with no churn to the underlying global routing system. [RFC7215] discusses the impact of LISP on the global routing system during the transition period. However, the separation between location and identity that LISP offers makes it suitable for use in additional scenarios such as Traffic Engineering (TE), multihoming, and mobility among others.

This document describes the LISP architecture and its main operational mechanisms as well as its design rationale. It is important to note that this document does not specify or complement the LISP protocol. The interested reader should refer to the main LISP specifications [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis], as well as the complementary documents [RFC6831], [RFC6832], [I-D.ietf-lisp-6834bis], [RFC6835], [RFC6836], [RFC7052] for the protocol specifications along with the LISP deployment guidelines [RFC7215].

2. Definition of Terms

Endpoint Identifier (EID): EIDs are addresses used to uniquely identify nodes irrespective of their topological location and are typically routed intra-domain.

Routing Locator (RLOC): RLOCs are addresses assigned topologically to network attachment points and typically routed inter-domain.

Ingress Tunnel Router (ITR): A LISP-capable router that encapsulates packets from a LISP site towards the core network.

Egress Tunnel Router (ETR): A LISP-capable router that decapsulates packets from the core of the network towards a LISP site.

xTR: A router that implements both ITR and ETR functionalities.

Map-Request: A LISP signaling message used to request an EID-to-RLOC mapping.

Map-Reply: A LISP signaling message sent in response to a Map-Request that contains a resolved EID-to-RLOC mapping.

Map-Register: A LISP signaling message used to register an EID-to-RLOC mapping.

Map-Notify: A LISP signaling message sent in response of a Map-Register to acknowledge the correct reception of an EID-to-RLOC mapping.

This document describes the LISP architecture and does not introduce any new term. The reader is referred to [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis], [RFC6831], [RFC6832], [I-D.ietf-lisp-6834bis], [RFC6835], [RFC6836], [RFC7052], [RFC7215] for the complete definition of terms.

3. LISP Architecture

This section presents the LISP architecture, it first details the design principles of LISP and then it proceeds to describe its main aspects: data-plane, control-plane, and internetworking mechanisms.

3.1. Design Principles

The LISP architecture is built on top of four basic design principles:

- * **Locator/Identifier split:** By decoupling the overloaded semantics of the current IP addresses the Internet core can be assigned identity meaningful addresses and hence, can use aggregation to scale. Devices are assigned with relatively opaque topologically meaningful addresses that are independent of their topological location.
- * **Overlay architecture:** Overlays route packets over the current Internet, allowing deployment of new protocols without changing the current infrastructure hence, resulting into a low deployment cost.
- * **Decoupled data-plane and control-plane:** Separating the data-plane from the control-plane allows them to scale independently and use different architectural approaches. This is important given that they typically have different requirements and allows for other data-planes to be added. Even though the data-plane and the control-plane are decoupled, they are not completely isolated because the LISP data-plane may trigger control-plane activity.
- * **Incremental deployability:** This principle ensures that the protocol interoperates with the legacy Internet while providing some of the targeted benefits to early adopters.

3.2. Overview of the Architecture

LISP splits architecturally the core from the edge of the Internet by creating two separate namespaces: Endpoint Identifiers (EIDs) and Routing LOCators (RLOCs). The edge consists of LISP sites (e.g., an Autonomous System) that use EID addresses. EIDs are IPv4 or IPv6 addresses that uniquely identify communication end-hosts and are assigned and configured by the same mechanisms that exist at the time of this writing. EIDs do not contain inter-domain topological information and because of this, EIDs are usually routable at the edge (within LISP sites) but not in the core; see Section 3.5 for discussion of LISP site internetworking with non-LISP sites and domains in the Internet.

LISP sites (at the edge) are connected to the interconnecting core by means of LISP-capable routers (e.g., border routers). LISP sites are connected across the interconnecting core using tunnels between the LISP-capable routers. When packets originated from a LISP site are flowing towards the core network, they ingress into an encapsulated tunnel via an Ingress Tunnel Router (ITR). When packets flow from the core network to a LISP site, they egress from an encapsulated tunnel to an Egress Tunnel Router (ETR). An xTR is a router which can perform both ITR and ETR operations. In this context ITRs encapsulate packets while ETRs decapsulate them, hence LISP operates as an overlay on top of the current Internet core.

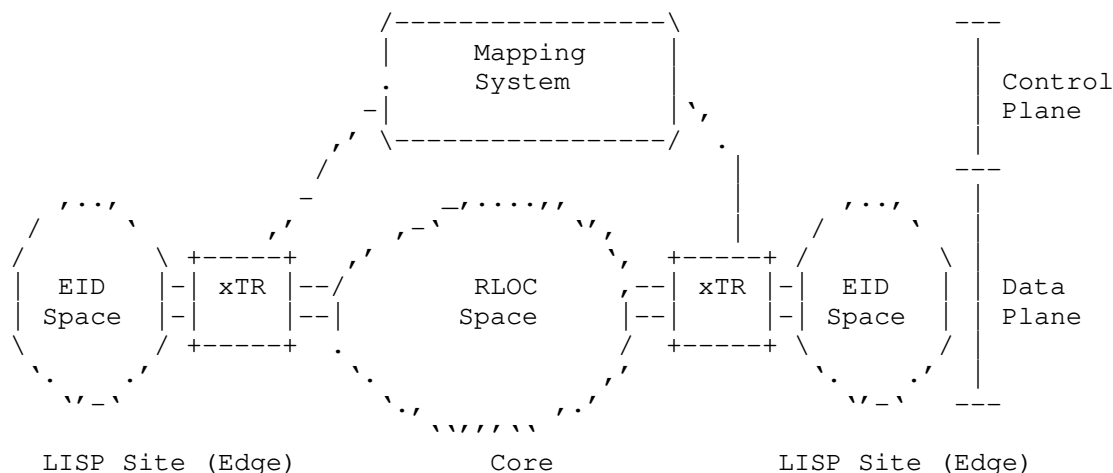


Figure 1: A schema of the LISP Architecture.

With LISP, the core uses RLOCs, an RLOC is an IPv4 or IPv6 address assigned to an core-facing network interface of an ITR or ETR.

A database which is typically distributed, called the Mapping System, stores mappings between EIDs and RLOCs. Such mappings relate the identity of the devices attached to LISP sites (EIDs) to the set of RLOCs configured at the LISP-capable routers servicing the site. Furthermore, the mappings also include traffic engineering policies and can be configured to achieve multihoming and load balancing. The LISP Mapping System is conceptually similar to the DNS where it is organized as a distributed multi-organization network database. With LISP, ETRs register mappings while ITRs retrieve them.

Finally, the LISP architecture emphasizes incremental deployment. Given that LISP represents an overlay to the current Internet architecture, end hosts as well as intra and inter-domain routers remain unchanged, and the only required changes to the existing infrastructure are to routers connecting the EID space with the RLOC space. Additionally, LISP requires the deployment of an independent Mapping System, such distributed database is a new network entity.

The following describes a simplified packet flow sequence between two nodes that are attached to LISP sites. Please note that typical LISP-capable routers are xTRs (both ITR and ETR). Client HostA wants to send a packet to server HostB.

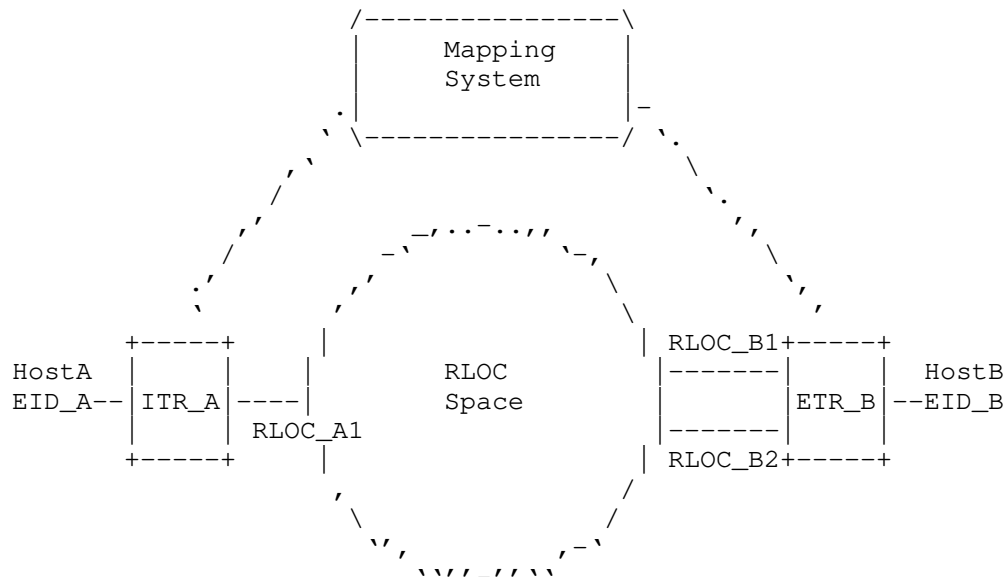


Figure 2: Packet flow sequence in LISP.

1. HostA retrieves the EID_B of HostB, typically querying the DNS and obtaining an A or AAAA record. Then it generates an IP packet as in the Internet, the packet has source address EID_A and destination address EID_B.
2. The packet is forwarded towards ITR_A in the LISP site using standard intra-domain mechanisms.
3. ITR_A upon receiving the packet queries the Mapping System to retrieve the locator of ETR_B that is servicing HostB's EID_B. In order to do so it uses a LISP control message called Map-Request, the message contains EID_B as the lookup key. In turn it receives another LISP control message called Map-Reply, the message contains two locators: RLOC_B1 and RLOC_B2 along with traffic engineering policies: priority and weight per locator. Note that a Map-Reply can contain more locators if needed. ITR_A can cache the mapping in a local storage to speed-up forwarding of subsequent packets.
4. ITR_A encapsulates the packet towards RLOC_B1 (chosen according to the priorities/weights specified in the mapping). The packet contains two IP headers, the outer header has RLOC_A1 as source and RLOC_B1 as destination, the inner original header has EID_A as source and EID_B as destination. Furthermore ITR_A adds a LISP header, more details about LISP encapsulation can be found in Section 3.3.1.
5. The encapsulated packet is forwarded over the interconnecting core as a normal IP packet, making the EID invisible from the core.
6. Upon reception of the encapsulated packet by ETR_B, it decapsulates the packet and forwards it to HostB.

3.3. Data-Plane

This section provides a high-level description of the LISP data-plane, which is specified in detail in [I-D.ietf-lisp-rfc6830bis]. The LISP data-plane is responsible for encapsulating and decapsulating data packets and caching the appropriate forwarding state. It includes two main entities, the ITR and the ETR, both are LISP capable routers that connect the EID with the RLOC space (ITR) and vice versa (ETR).

3.3.1. LISP Encapsulation

ITRs encapsulate data packets towards ETRs. LISP data packets are encapsulated using UDP (port 4341), the source port is usually selected by the ITR using a 5-tuple hash of the inner header (so to be consistent in case of multi-path solutions such as ECMP [RFC2992]) and ignored on reception. LISP data packets are often encapsulated in UDP packets that include a zero checksum [RFC6935] [RFC6936] that may not be verified when it is received, because LISP data packets typically include an inner transport protocol header with a non-zero checksum. The use of UDP zero checksums over IPv6 for all tunneling protocols like LISP is subject to the applicability statement in [RFC6936]. If LISP data packets are encapsulated in UDP packets with non-zero checksums, the outer UDP checksums are verified when the UDP packets are received, as part of normal UDP processing.

LISP-encapsulated packets also include a LISP header (after the UDP header and before the original IP header). The LISP header is prepended by ITRs and stripped by ETRs. It carries reachability information (see more details in Section 4.2) and the Instance ID field. The Instance ID field is used to distinguish traffic to/from different tenant address spaces at the LISP site and that may use overlapped but logically separated EID addressing.

Overall, LISP works on 4 headers, the inner header the source constructed, and the 3 headers a LISP encapsulator prepends ("outer" to "inner"):

1. Outer IP header containing RLOCs as source and destination addresses. This header is originated by ITRs and stripped by ETRs.
2. UDP header (port 4341), usually with zero checksum. This header is originated by ITRs and stripped by ETRs.
3. LISP header that contains various forwarding-plane features (such as reachability) and an Instance ID field. This header is originated by ITRs and stripped by ETRs.
4. Inner IP header containing EIDs as source and destination addresses. This header is created by the source end-host and is left unchanged by LISP data plane processing on the ITR and ETR.

Finally, in some scenarios Re-encapsulating and/or Recursive tunnels are useful to choose a specified path in the underlay network, for instance to avoid congestion or failure. Re-encapsulating tunnels are consecutive LISP tunnels and occur when a decapsulator (an ETR action) removes a LISP header and then acts as an encapsulator (an ITR

action) to prepend another one. On the other hand, Recursive tunnels are nested tunnels and are implemented by using multiple LISP encapsulations on a packet. Such functions are implemented by Reencapsulating Tunnel Routers (RTRs). An RTR can be thought of as a router that first acts as an ETR by decapsulating packets and then as an ITR by encapsulating them towards another locator, more information can be found at [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis].

3.3.2. LISP Forwarding State

In the LISP architecture, ITRs keep just enough information to route traffic flowing through them. Meaning that, ITRs retrieve from the LISP Mapping System mappings between EID-prefixes (blocks of EIDs) and RLOCs that are used to encapsulate packets. Such mappings are stored in a local cache called the LISP Map-Cache for subsequent packets addressed to the same EID prefix. Note that, in case of overlapping EID-prefixes, following a single request, the ITR may receive a set of mappings, covering the requested EID-prefix and all more-specifics (cf., Section 5.5 [I-D.ietf-lisp-rfc6833bis]). Mappings include a (Time-to-Live) TTL (set by the ETR). More details about the Map-Cache management can be found in Section 4.1.

3.4. Control-Plane

The LISP control-plane, specified in [I-D.ietf-lisp-rfc6833bis], provides a standard interface to register and request mappings. The LISP Mapping System is a database that stores such mappings. The following first describes the mappings, then the standard interface to the Mapping System, and finally its architecture.

3.4.1. LISP Mappings

Each mapping includes the bindings between EID prefix(es) and set of RLOCs as well as traffic engineering policies, in the form of priorities and weights for the RLOCs. Priorities allow the ETR to configure active/backup policies while weights are used to load-balance traffic among the RLOCs (on a per-flow basis).

Typical mappings in LISP bind EIDs in the form of IP prefixes with a set of RLOCs, also in the form of IP addresses. IPv4 and IPv6 addresses are encoded using the appropriate Address Family Identifier (AFI) [RFC3232]. However LISP can also support more general address encoding by means of the ongoing effort around the LISP Canonical Address Format (LCAF) [RFC8060].

With such a general syntax for address encoding in place, LISP aims to provide flexibility to current and future applications. For instance LCAFs could support MAC addresses, geo-coordinates, ASCII names and application specific data.

3.4.2. Mapping System Interface

LISP defines a standard interface between data and control planes. The interface is specified in [I-D.ietf-lisp-rfc6833bis] and defines two entities:

Map-Server: A network infrastructure component that learns mappings from ETRs and publishes them into the LISP Mapping System. Typically Map-Servers are not authoritative to reply to queries and hence, they forward them to the ETR. However, they can also operate in proxy-mode, where the ETRs delegate replying to queries to Map-Servers. This setup is useful when the ETR has limited resources (e.g., CPU or power).

Map-Resolver: A network infrastructure component that interfaces ITRs with the Mapping System by proxying queries and in some cases responses.

The interface defines four LISP control messages which are sent as UDP datagrams (port 4342):

Map-Register: This message is used by ETRs to register mappings in the Mapping System and it is authenticated using a shared key between the ETR and the Map-Server.

Map-Notify: When requested by the ETR, this message is sent by the Map-Server in response to a Map-Register to acknowledge the correct reception of the mapping and convey the latest Map-Server state on the EID to RLOC mapping. In some cases a Map-Notify can be sent to the previous RLOCs when an EID is registered by a new set of RLOCs.

Map-Request: This message is used by ITRs or Map-Resolvers to resolve the mapping of a given EID.

Map-Reply: This message is sent by Map-Servers or ETRs in response to a Map-Request and contains the resolved mapping. Please note that a Map-Reply may contain a negative reply if, for example, the queried EID is not part of the LISP EID space. In such cases the ITR typically forwards the traffic natively (non encapsulated) to the public Internet, this behavior is defined to support incremental deployment of LISP.

3.4.3. Mapping System

LISP architecturally decouples control and data-plane by means of a standard interface. This interface glues the data-plane - routers responsible for forwarding data-packets - with the LISP Mapping System - a database responsible for storing mappings.

With this separation in place, the data and control-plane can use different architectures if needed and scale independently. Typically the data-plane is optimized to route packets according to hierarchical IP addresses. However the control-plane may have different requirements, for instance and by taking advantage of the LCAF, the Mapping System may be used to store non-hierarchical keys (such as MAC addresses), requiring different architectural approaches for scalability. Another important difference between the LISP control- and data- planes is that, as a result of the local mapping cache available at ITR, the Mapping System does not need to operate at line-rate.

Many of the existing mechanisms to create distributed systems have been explored and considered for the Mapping System architecture: graph-based databases in the form of LISP+ALT [RFC6836], hierarchical databases in the form of LISP-DDT [RFC8111], monolithic databases in the form of LISP-NERD [RFC6837], flat databases in the form of LISP-DHT [I-D.cheng-lisp-shdht], [Mathy], and a multicast-based database [I-D.curran-lisp-emacs]. Furthermore it is worth noting that, in some scenarios such as private deployments, the Mapping System can operate as logically centralized. In such cases it is typically composed of a single Map-Server/Map-Resolver.

The following focuses on the two mapping systems that have been implemented and deployed (LISP+ALT and LISP-DDT).

3.4.3.1. LISP+ALT

The LISP Alternative Topology (LISP+ALT) [RFC6836] was the first Mapping System proposed, developed and deployed on the LISP pilot network. It is based on a distributed BGP overlay participated by Map-Servers and Map-Resolvers. The nodes connect to their peers through static tunnels. Each Map-Server involved in the ALT topology advertises the EID-prefixes registered by the serviced ETRs, making the EID routable on the ALT topology.

When an ITR needs a mapping it sends a Map-Request to a Map-Resolver that, using the ALT topology, forwards the Map-Request towards the Map-Server responsible for the mapping. Upon reception the Map-Server forwards the request to the ETR that in turn, replies directly to the ITR.

3.4.3.2. LISP-DDT

LISP-DDT [RFC8111] is conceptually similar to the DNS, a hierarchical directory whose internal structure mirrors the hierarchical nature of the EID address space. The DDT hierarchy is composed of DDT nodes forming a tree structure, the leafs of the tree are Map-Servers. On top of the structure there is the DDT root node, which is a particular instance of a DDT node and that matches the entire address space. As in the case of DNS, DDT supports multiple redundant DDT nodes and/or DDT roots. Finally, Map-Resolvers are the clients of the DDT hierarchy and can query either the DDT root and/or other DDT nodes.

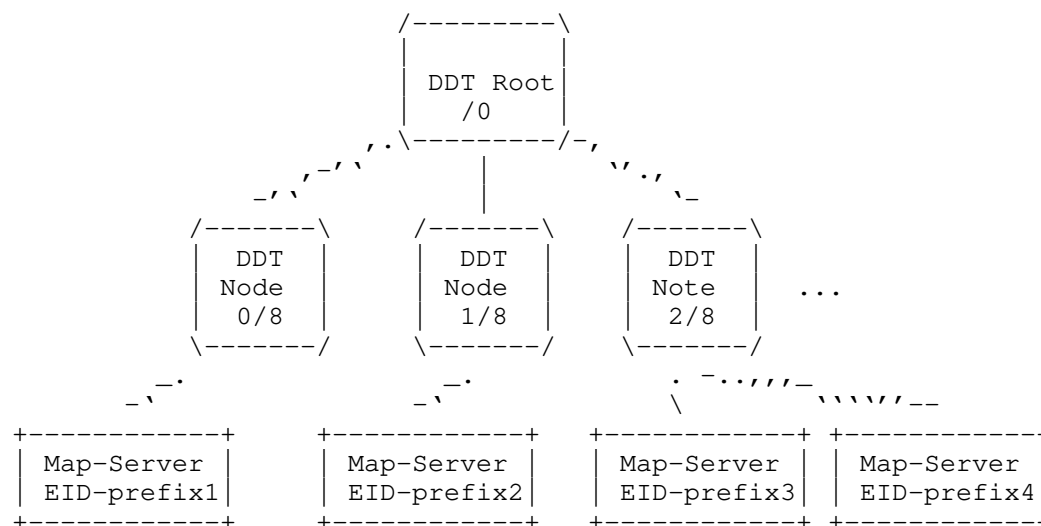


Figure 3: A schematic representation of the DDT tree structure, please note that the prefixes and the structure depicted should be only considered as an example.

The DDT structure does not actually index EID-prefixes but eXtended EID-prefixes (XEID). An XEID-prefix is just the concatenation of the following fields (from most significant bit to less significant bit): Database-ID, Instance ID, Address Family Identifier and the actual EID-prefix. The Database-ID is provided for possible future requirements of higher levels in the hierarchy and to enable the creation of multiple and separate database trees.

In order to resolve a query LISP-DDT operates in a similar way to the DNS but only supports iterative lookups. DDT clients (usually Map-Resolvers) generate Map-Requests to the DDT root node. In response

they receive a newly introduced LISP-control message: a Map-Referral. A Map-Referral provides the list of RLOCs of the set of DDT nodes matching a configured XEID delegation. That is, the information contained in the Map-Referral points to the child of the queried DDT node that has more specific information about the queried XEID-prefix. This process is repeated until the DDT client walks the tree structure (downwards) and discovers the Map-Server servicing the queried XEID. At this point the client sends a Map-Request and receives a Map-Reply containing the mappings. It is important to note that DDT clients can also cache the information contained in Map-Referrals, that is, they cache the DDT structure. This is used to reduce the mapping retrieving latency [Jakab].

The DDT Mapping System relies on manual configuration. That is Map-Resolvers are configured with the set of available DDT root nodes while DDT nodes are configured with the appropriate XEID delegations. Configuration changes in the DDT nodes are only required when the tree structure changes itself, but it doesn't depend on EID dynamics (RLOC allocation or traffic engineering policy changes).

3.5. Internetworking Mechanisms

EIDs are typically identical to either IPv4 or IPv6 addresses and they are stored in the LISP Mapping System, however they are usually not announced in the routing system beyond the local LISP domain. As a result LISP requires an internetworking mechanism to allow LISP sites to speak with non-LISP sites and vice versa. LISP internetworking mechanisms are specified in [RFC6832].

LISP defines two entities to provide internetworking:

Proxy Ingress Tunnel Router (PITR): PITRs provide connectivity from the legacy Internet to LISP sites. PITRs announce in the global routing system blocks of EID prefixes (aggregating when possible) to attract traffic. For each incoming packet from a source not in a LISP site (a non-EID), the PITR LISP-encapsulates it towards the RLOC(s) of the appropriate LISP site. The impact of PITRs in the routing table size of the Default-Free Zone (DFZ) is, in the worst-case, similar to the case in which LISP is not deployed. EID-prefixes will be aggregated as much as possible both by the PITR and by the global routing system.

Proxy Egress Tunnel Router (PETR): PETRs provide connectivity from LISP sites to the legacy Internet. In some scenarios, LISP sites may be unable to send encapsulated packets with a local EID address as a source to the legacy Internet. For instance when Unicast Reverse Path Forwarding (uRPF) is used by Provider Edge routers, or when an intermediate network between a LISP site and a

non-LISP site does not support the desired version of IP (IPv4 or IPv6). In both cases the PETR overcomes such limitations by encapsulating packets over the network. There is no specified provision for the distribution of PETR RLOC addresses to the ITRs.

Additionally, LISP also defines mechanisms to operate with private EIDs [RFC1918] by means of LISP-NAT [RFC6832]. In this case the xTR replaces a private EID source address with a routable one. At the time of this writing, work is ongoing to define NAT-traversal capabilities, that is xTRs behind a NAT using non-routable RLOCs.

PITRs, PETRs and, LISP-NAT enable incremental deployment of LISP, by providing significant flexibility in the placement of the boundaries between the LISP and non-LISP portions of the network, and making it easy to change those boundaries over time.

4. LISP Operational Mechanisms

This section details the main operational mechanisms defined in LISP.

4.1. Cache Management

LISP's decoupled control and data-plane, where mappings are stored in the control-plane and used for forwarding in the data-plane, requires a local cache in ITRs to reduce signaling overhead (Map-Request/Map-Reply) and increase forwarding speed. The local cache available at the ITRs, called Map-Cache, is used by the router to LISP-encapsulate packets. The Map-Cache is indexed by (Instance ID, EID-prefix) and contains basically the set of RLOCs with the associated traffic engineering policies (priorities and weights).

The Map-Cache, as any other cache, requires cache coherence mechanisms to maintain up-to-date information. LISP defines three main mechanisms for cache coherence:

Record Time-To-Live (TTL): Each mapping record contains a TTL set by the ETR, upon expiration of the TTL the ITR can't use the mapping until it is refreshed by sending a new Map-Request.

Solicit-Map-Request (SMR): SMR is an explicit mechanism to update mapping information. In particular a special type of Map-Request can be sent on demand by ETRs to request refreshing a mapping. Upon reception of a SMR message, the ITR must refresh the bindings by sending a Map-Request to the Mapping System. Further uses of SMRs are documented in [I-D.ietf-lisp-rfc6833bis].

Map-Versioning: This optional mechanism piggybacks in the LISP

header of data-packets the version number of the mappings used by an xTR. This way, when an xTR receives a LISP-encapsulated packet from a remote xTR, it can check whether its own Map-Cache or the one of the remote xTR is outdated. If its Map-Cache is outdated, it sends a Map-Request for the remote EID so to obtain the newest mappings. On the contrary, if it detects that the remote xTR Map-Cache is outdated, it sends a SMR to notify it that a new mapping is available. Further details are available in [I-D.ietf-lisp-6834bis].

Finally it is worth noting that in some cases an entry in the map-cache can be proactively refreshed using the mechanisms described in the section below.

4.2. RLOC Reachability

In most cases LISP operates with a pull-based Mapping System (e.g., DDT), this results in an edge to edge pull architecture. In such scenario the network state is stored in the control-plane while the data-plane pulls it on demand. This has consequences concerning the propagation of xTRs reachability/liveness information since pull architectures require explicit mechanisms to propagate this information. As a result LISP defines a set of mechanisms to inform ITRs and PITRs about the reachability of the cached RLOCs:

Locator Status Bits (LSB): LSB is a passive technique, the LSB field is carried by data-packets in the LISP header and can be set by a ETRs to specify which RLOCs of the ETR site are up/down. This information can be used by the ITRs as a hint about the reachability to perform additional checks. Also note that LSB does not provide path reachability status, only hints on the status of RLOCs as such they must not be used over the public Internet and should be coupled with Map-Versioning to prevent race conditions where LSB are interpreted as referring to different RLOCs than intended.

Echo-nonce: This is also a passive technique, that can only operate effectively when data flows bi-directionally between two communicating xTRs. Basically, an ITR piggybacks a random number (called nonce) in LISP data packets, if the path and the probed locator are up, the ETR will piggyback the same random number on the next data-packet, if this is not the case the ITR can set the locator as unreachable. When traffic flow is unidirectional or when the ETR receiving the traffic is not the same as the ITR that transmits it back, additional mechanisms are required. The echo-nonce mechanism must be used in trusted environments only, not over the public Internet.

RLOC-probing: This is an active probing algorithm where ITRs send probes to specific locators, this effectively probes both the locator and the path. In particular this is done by sending a Map-Request (with certain flags activated) on the data-plane (RLOC space) and waiting in return a Map-Reply, also sent on the data-plane. The active nature of RLOC-probing provides an effective mechanism to determine reachability and, in case of failure, switching to a different locator. Furthermore the mechanism also provides useful RTT estimates of the delay of the path that can be used by other network algorithms.

It is worth noting that RLOC probing and Echo-nonce can work together. Specifically if a nonce is not echoed, an ITR could RLOC-probe to determine if the path is up when it cannot tell the difference between a failed bidirectional path or the return path is not used (a unidirectional path).

Additionally, LISP also recommends inferring reachability of locators by using information provided by the underlay, in particular:

ICMP signaling: The LISP underlay -the current Internet- uses the ICMP protocol to signal unreachability (among other things). LISP can take advantage of this and the reception of a ICMP Network Unreachable or ICMP Host Unreachable message can be seen as a hint that a locator might be unreachable, this should lead to perform additional checks.

Underlay routing: Both BGP and IGP carry reachability information, LISP-capable routers that have access to underlay routing information can use it to determine if a given locator or path are reachable.

4.3. ETR Synchronization

All the ETRs that are authoritative to a particular EID-prefix must announce the same mapping to the requesters, this means that ETRs must be aware of the status of the RLOCs of the remaining ETRs. This is known as ETR synchronization.

At the time of this writing LISP does not specify a mechanism to achieve ETR synchronization. Although many well-known techniques could be applied to solve this issue it is still under research, as a result operators must rely on coherent manual configuration

4.4. MTU Handling

Since LISP encapsulates packets it requires dealing with packets that exceed the MTU of the path between the ITR and the ETR. Specifically LISP defines two mechanisms:

Stateless: With this mechanism the effective MTU is assumed from the ITR's perspective. If a payload packet is too big for the effective MTU, and can be fragmented, the payload packet is fragmented on the ITR, such that reassembly is performed at the destination host.

Stateful: With this mechanism ITRs keep track of the MTU of the paths towards the destination locators by parsing the ICMP Too Big packets sent by intermediate routers. ITRs will send ICMP Too Big messages to inform the sources about the effective MTU. Additionally ITRs can use mechanisms such as PMTUD [RFC1191] or PLPMTUD [RFC4821] to keep track of the MTU towards the locators.

In both cases if the packet cannot be fragmented (IPv4 with DF=1 or IPv6) then the ITR drops it and replies with a ICMP Too Big message to the source.

5. Mobility

The separation between locators and identifiers in LISP is suitable for traffic engineering purpose where LISP sites can change their attachment points to the Internet (i.e., RLOCs) without impacting endpoints or the Internet core. In this context, the border routers operate the xTR functionality and endpoints are not aware of the existence of LISP. This functionality is similar to Network Mobility [RFC3963]. However, this mode of operation does not allow seamless mobility of endpoints between different LISP sites as the EID address might not be routable in a visited site. Nevertheless, LISP can be used to enable seamless IP mobility when LISP is directly implemented in the endpoint or when the endpoint roams to an attached xTR. Each endpoint is then an xTR and the EID address is the one presented to the network stack used by applications while the RLOC is the address gathered from the network when it is visited. This functionality is similar to Mobile IP ([RFC5944] and [RFC6275]).

Whenever the device changes of RLOC, the xTR updates the RLOC of its local mapping and registers it to its Map-Server, typically with a low TTL value (1min). To avoid the need of a home gateway, the ITR also indicates the RLOC change to all remote devices that have ongoing communications with the device that moved. The combination of both methods ensures the scalability of the system as signaling is strictly limited the Map-Server and to hosts with which communications are ongoing. In the mobility case the EID-prefix can be as small as a full /32 or /128 (IPv4 or IPv6 respectively) depending on the specific use-case (e.g., subnet mobility vs single VM/Mobile node mobility).

The decoupled identity and location provided by LISP allows it to operate with other layer 2 and layer 3 mobility solutions.

6. Multicast

LISP also supports transporting IP multicast packets sent from the EID space, the operational changes required to the multicast protocols are documented in [RFC6831].

In such scenarios, LISP may create multicast state both at the core and at the sites (both source and receiver). When signaling is used to create multicast state at the sites, LISP routers unicast encapsulate PIM Join/Prune messages from receiver to source sites. At the core, ETRs build a new PIM Join/Prune message addressed to the RLOC of the ITR servicing the source. An simplified sequence is shown below

1. An end-host willing to join a multicast channel sends an IGMP report. Multicast PIM routers at the LISP site propagate PIM Join/Prune messages (S-EID, G) towards the ETR.
2. The join message flows to the ETR, upon reception the ETR builds two join messages, the first one unicast LISP-encapsulates the original join message towards the RLOC of the ITR servicing the source. This message creates (S-EID, G) multicast state at the source site. The second join message contains as destination address the RLOC of the ITR servicing the source (S-RLOC, G) and creates multicast state at the core.
3. Multicast data packets originated by the source (S-EID, G) flow from the source to the ITR. The ITR LISP-encapsulates the multicast packets, the outer header includes its own RLOC as the source (S-RLOC) and the original multicast group address (G) as the destination. Please note that multicast group address are logical and are not resolved by the mapping system. Then the multicast packet is transmitted through the core towards the receiving ETRs that decapsulates the packets and sends them using the receiver's site multicast state.

Please note that the inner and outer multicast addresses are in general different, unless in specific cases where the underlay provider implements a tight control on the overlay. LISP specifications already support all PIM modes [RFC6831]. Additionally, LISP can support as well non-PIM mechanisms in order to maintain multicast state.

When multicast sources and receivers are active at LISP sites, and the core network between the sites does not provide multicast support, a signal-free mechanism can be used to create an overlay that will allow multicast traffic to flow between sites and connect the multicast trees at the different sites [RFC8378]. Registrations from the different receiver sites will be merged at the mapping system to assemble a multicast-replication-list inclusive of all Routing Locators (RLOCs) that lead to receivers for a particular multicast group or multicast channel. The replication list for each specific multicast entry is maintained as a database mapping entry in the LISP mapping system.

7. Use Cases

7.1. Traffic Engineering

A LISP site can strictly impose via which ETRs the traffic must enter the LISP site network even though the path followed to reach the ETR is not under the control of the LISP site. This fine control is implemented with the mappings. When a remote site is willing to send traffic to a LISP site, it retrieves the mapping associated to the destination EID via the mapping system. The mapping is sent directly by an authoritative ETR of the EID and is not altered by any intermediate network.

A mapping associates a list of RLOCs to an EID prefix. Each RLOC corresponds to an interface of an ETR (or set of ETRs) that is able to correctly forward packets to EIDs in the prefix. Each RLOC is tagged with a priority and a weight in the mapping. The priority is used to indicate which RLOCs should be preferred to send packets (the least preferred ones being provided for backup purpose). The weight permits to balance the load between the RLOCs with the same priority, proportionally to the weight value.

As mappings are directly issued by the authoritative ETR of the EID and are not altered while transmitted to the remote site, it offers highly flexible incoming inter-domain traffic engineering with even the possibility for a site to support a different mapping policy for each remote site.

7.2. LISP for IPv6 Co-existence

LISP encapsulations allow to transport packets using EIDs from a given address family (e.g., IPv6) with packets from other address families (e.g., IPv4). The absence of correlation between the address family of RLOCs and EIDs makes LISP a candidate to allow, e.g., IPv6 to be deployed when all of the core network may not have IPv6 enabled.

For example, two IPv6-only data centers could be interconnected via the legacy IPv4 Internet. If their border routers are LISP capable, sending packets between the data center is done without any form of translation as the native IPv6 packets (in the EID space) will be LISP encapsulated and transmitted over the IPv4 legacy Internet by the mean of IPv4 RLOCs.

7.3. LISP for Virtual Private Networks

It is common to operate several virtual networks over the same physical infrastructure. In such virtual private networks, it is essential to distinguish which virtual network a packet belongs and tags or labels are used for that purpose. When using LISP, the distinction can be made with the Instance ID field. When an ITR encapsulates a packet from a particular virtual network (e.g., known via the VRF or VLAN), it tags the encapsulated packet with the Instance ID corresponding to the virtual network of the packet. When an ETR receives a packet tagged with an Instance ID it uses the Instance ID to determine how to treat the packet.

The main usage of LISP for virtual private networks does not introduce additional requirements on the underlying network, as long as it runs IP.

7.4. LISP for Virtual Machine Mobility in Data Centers

A way to enable seamless virtual machine mobility in data center is to conceive the datacenter backbone as the RLOC space and the subnet where servers are hosted as forming the EID space. A LISP router is placed at the border between the backbone and each subnet. When a virtual machine is moved to another subnet, it can keep (temporarily) the address it had before the move so to continue without a transport layer connection reset. When an xTR detects a source address received on a subnet to be an address not assigned to the subnet, it registers the address to the Mapping System.

To inform the other LISP routers that the machine moved and where, and then to avoid detours via the initial subnetwork, mechanisms such as the Solicit-Map-Request messages are used.

8. Security Considerations

This section describes the security considerations associated to the LISP protocol.

While in a push mapping system, the state necessary to forward packets is learned independently of the traffic itself, with a pull architecture, the system becomes reactive and data-plane events

(e.g., the arrival of a packet for an unknown destination) may trigger control-plane events. This on-demand learning of mappings provides many advantages as discussed above but may also affect the way security is enforced.

Usually, the data-plane is implemented in the fast path of routers to provide high performance forwarding capabilities while the control-plane features are implemented in the slow path to offer high flexibility and a performance gap of several order of magnitude can be observed between the slow and the fast paths. As a consequence, the way data-plane events are notified to the control-plane must be thought carefully so to not overload the slow path and rate limiting should be used as specified in [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis].

Care must also be taken so to not overload the mapping system (i.e., the control plane infrastructure) as the operations to be performed by the mapping system may be more complex than those on the data-plane, for that reason [I-D.ietf-lisp-rfc6830bis] and [I-D.ietf-lisp-rfc6833bis] recommends to rate limit the sending of messages to the mapping system.

To improve resiliency and reduce the overall number of messages exchanged, LISP offers the possibility to leak information, such as reachability of locators, directly into data plane packets. In environments that are not fully trusted, like the open Internet, control information gleaned from data-plane packets must not be used or must be verified before using it.

Mappings are the centrepiece of LISP and all precautions must be taken to avoid them to be manipulated or misused by malicious entities. Using trustable Map-Servers that strictly respect [I-D.ietf-lisp-rfc6833bis] and the authentication mechanism proposed by LISP-Sec [I-D.ietf-lisp-sec] reduces the risk of attacks to the mapping integrity. In more critical environments, secure measures may be needed. The way security is implemented for a given mapping system strongly depends on the architecture of the mapping system itself and the threat model assumed for the deployment. Thus, the mapping system security has to be discussed in the relevant documents proposing the mapping system architecture.

As with any other tunneling mechanism, middleboxes on the path between an ITR (or PITR) and an ETR (or PETR) must implement mechanisms to strip the LISP encapsulation to correctly inspect the content of LISP encapsulated packets.

Like other map-and-encap mechanisms, LISP enables triangular routing (i.e., packets of a flow cross different border routers depending on their direction). This means that intermediate boxes may have incomplete view on the traffic they inspect or manipulate. Moreover, LISP-encapsulated packets are routed based on the outer IP address (i.e., the RLOC), and can be delivered to an ETR that is not responsible of the destination EID of the packet or even to a network element that is not an ETR. The mitigation consists in applying appropriate filtering techniques on the network elements that can potentially receive un-expected LISP-encapsulated packets

More details about security implications of LISP are discussed in [RFC7835].

9. IANA Considerations

This memo includes no requests to IANA.

10. Acknowledgements

This document was initiated by Noel Chiappa and much of the core philosophy came from him. The authors acknowledge the important contributions he has made to this work and thank him for his past efforts.

The authors would also like to thank Dino Farinacci, Fabio Maino, Luigi Iannone, Sharon Barkai, Isidoros Kouvelas, Christian Cassar, Florin Coras, Marc Binderberger, Alberto Rodriguez-Natal, Ronald Bonica, Chad Hintz, Robert Raszuk, Joel M. Halpern, Darrel Lewis, David Black.

11. References

11.1. Normative References

[I-D.ietf-lisp-6834bis]
Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", Work in Progress, Internet-Draft, draft-ietf-lisp-6834bis-09, 31 August 2021, <<https://www.ietf.org/archive/id/draft-ietf-lisp-6834bis-09.txt>>.

- [I-D.ietf-lisp-rfc6830bis]
Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos-Aparicio, "The Locator/ID Separation Protocol (LISP)", Work in Progress, Internet-Draft, draft-ietf-lisp-rfc6830bis-36, 18 November 2020, <<https://www.ietf.org/internet-drafts/draft-ietf-lisp-rfc6830bis-36.txt>>.
- [I-D.ietf-lisp-rfc6833bis]
Farinacci, D., Maino, F., Fuller, V., and A. Cabellos-Aparicio, "Locator/ID Separation Protocol (LISP) Control-Plane", Work in Progress, Internet-Draft, draft-ietf-lisp-rfc6833bis-30, 18 November 2020, <<https://www.ietf.org/internet-drafts/draft-ietf-lisp-rfc6833bis-30.txt>>.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", Work in Progress, Internet-Draft, draft-ietf-lisp-sec-22, 12 January 2021, <<https://www.ietf.org/internet-drafts/draft-ietf-lisp-sec-22.txt>>.
- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <<https://www.rfc-editor.org/info/rfc1191>>.
- [RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <<https://www.rfc-editor.org/info/rfc1918>>.
- [RFC2992] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", RFC 2992, DOI 10.17487/RFC2992, November 2000, <<https://www.rfc-editor.org/info/rfc2992>>.
- [RFC3232] Reynolds, J., Ed., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, DOI 10.17487/RFC3232, January 2002, <<https://www.rfc-editor.org/info/rfc3232>>.
- [RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, DOI 10.17487/RFC3963, January 2005, <<https://www.rfc-editor.org/info/rfc3963>>.
- [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <<https://www.rfc-editor.org/info/rfc4821>>.

- [RFC4984] Meyer, D., Ed., Zhang, L., Ed., and K. Fall, Ed., "Report from the IAB Workshop on Routing and Addressing", RFC 4984, DOI 10.17487/RFC4984, September 2007, <<https://www.rfc-editor.org/info/rfc4984>>.
- [RFC5944] Perkins, C., Ed., "IP Mobility Support for IPv4, Revised", RFC 5944, DOI 10.17487/RFC5944, November 2010, <<https://www.rfc-editor.org/info/rfc5944>>.
- [RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <<https://www.rfc-editor.org/info/rfc6275>>.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, DOI 10.17487/RFC6831, January 2013, <<https://www.rfc-editor.org/info/rfc6831>>.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, DOI 10.17487/RFC6832, January 2013, <<https://www.rfc-editor.org/info/rfc6832>>.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, DOI 10.17487/RFC6835, January 2013, <<https://www.rfc-editor.org/info/rfc6835>>.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, DOI 10.17487/RFC6836, January 2013, <<https://www.rfc-editor.org/info/rfc6836>>.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, DOI 10.17487/RFC6837, January 2013, <<https://www.rfc-editor.org/info/rfc6837>>.
- [RFC6935] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", RFC 6935, DOI 10.17487/RFC6935, April 2013, <<https://www.rfc-editor.org/info/rfc6935>>.
- [RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, DOI 10.17487/RFC6936, April 2013, <<https://www.rfc-editor.org/info/rfc6936>>.

- [RFC7052] Schudel, G., Jain, A., and V. Moreno, "Locator/ID Separation Protocol (LISP) MIB", RFC 7052, DOI 10.17487/RFC7052, October 2013, <<https://www.rfc-editor.org/info/rfc7052>>.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", RFC 7215, DOI 10.17487/RFC7215, April 2014, <<https://www.rfc-editor.org/info/rfc7215>>.
- [RFC7835] Saucez, D., Iannone, L., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Threat Analysis", RFC 7835, DOI 10.17487/RFC7835, April 2016, <<https://www.rfc-editor.org/info/rfc7835>>.
- [RFC8060] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", RFC 8060, DOI 10.17487/RFC8060, February 2017, <<https://www.rfc-editor.org/info/rfc8060>>.
- [RFC8111] Fuller, V., Lewis, D., Ermagan, V., Jain, A., and A. Smirnov, "Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)", RFC 8111, DOI 10.17487/RFC8111, May 2017, <<https://www.rfc-editor.org/info/rfc8111>>.
- [RFC8378] Moreno, V. and D. Farinacci, "Signal-Free Locator/ID Separation Protocol (LISP) Multicast", RFC 8378, DOI 10.17487/RFC8378, May 2018, <<https://www.rfc-editor.org/info/rfc8378>>.

11.2. Informative References

- [I-D.cheng-lisp-shdht]
Cheng, L. and J. Wang, "LISP Single-Hop DHT Mapping Overlay", Work in Progress, Internet-Draft, draft-cheng-lisp-shdht-04, 15 July 2013, <<http://www.ietf.org/internet-drafts/draft-cheng-lisp-shdht-04.txt>>.
- [I-D.curran-lisp-emacs]
Brim, S., Farinacci, D., Meyer, D., and J. Curran, "EID Mappings Multicast Across Cooperating Systems for LISP", Work in Progress, Internet-Draft, draft-curran-lisp-emacs-00, 9 November 2007, <<http://tools.ietf.org/html/draft-curran-lisp-emacs-00>>.

- [Jakab] Jakab, L., Cabellos, A., Saucez, D., and O. Bonaventure, "LISP-TREE: A DNS Hierarchy to Support the LISP Mapping System, IEEE Journal on Selected Areas in Communications, vol. 28, no. 8, pp. 1332-1343", October 2010.
- [Mathy] Mathy, L., Iannone, L., and O. Bonaventure, "LISP-DHT: Towards a DHT to map identifiers onto locators. The ACM ReArch, Re-Architecting the Internet. Madrid (Spain)", December 2008.
- [Quoitin] Quoitin, B., Iannone, L., Launois, C., and O. Bonaventure, "Evaluating the Benefits of the Locator/Identifier Separation" in Proceedings of 2Nd ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture", 2007.

Appendix A. A Brief History of Location/Identity Separation

The LISP architecture for separation of location and identity resulted from the discussions of this topic at the Amsterdam IAB Routing and Addressing Workshop, which took place in October 2006 [RFC4984].

A small group of like-minded personnel spontaneously formed immediately after that workshop, to work on an idea that came out of informal discussions at the workshop and on various mailing lists. The first Internet-Draft on LISP appeared in January, 2007.

Trial implementations started at that time, with initial trial deployments underway since June 2007; the results of early experience have been fed back into the design in a continuous, ongoing process over several years. LISP at this point represents a moderately mature system, having undergone a long organic series of changes and updates.

LISP transitioned from an IRTF activity to an IETF WG in March 2009, and after numerous revisions, the basic specifications moved to becoming RFCs at the start of 2013 (although work to expand and improve it, and find new uses for it, continues, and undoubtedly will for a long time to come). The LISP WG was rechartered in 2018 to continue work on the LISP base protocol and produce standard-track documents.

A.1. Old LISP Models

LISP, as initially conceived, had a number of potential operating modes, named 'models'. Although they are no used anymore, one occasionally sees mention of them, so they are briefly described here.

LISP 1: EIDs all appear in the normal routing and forwarding tables of the network (i.e. they are 'routable'); this property is used to 'bootstrap' operation, by using this to load EID->RLOC mappings. Packets were sent with the EID as the destination in the outer wrapper; when an ETR saw such a packet, it would send a Map-Reply to the source ITR, giving the full mapping.

LISP 1.5: Similar to LISP 1, but the routability of EIDs happens on a separate network.

LISP 2: EIDs are not routable; EID->RLOC mappings are available from the DNS.

LISP 3: EIDs are not routable; and have to be looked up in in a new EID->RLOC mapping database (in the initial concept, a system using Distributed Hash Tables). Two variants were possible: a 'push' system, in which all mappings were distributed to all ITRs, and a 'pull' system in which ITRs load the mappings they need, as needed.

Authors' Addresses

Albert Cabellos
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: acabello@ac.upc.edu

Damien Saucez (Ed.)
Inria
2004 route des Lucioles BP 93
06902 Sophia Antipolis Cedex
France

Email: damien.saucez@inria.fr

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 24, 2014

D. Saucez
INRIA
L. Iannone
Telecom ParisTech
O. Bonaventure
Universite catholique de Louvain
October 21, 2013

LISP Threats Analysis
draft-ietf-lisp-threats-08.txt

Abstract

This document proposes a threat analysis of the Locator/Identifier Separation Protocol (LISP) if deployed in the Internet.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 24, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. On-path Attackers	3
3. Off-Path Attackers: Reference Environment	4
4. Attack vectors	5
4.1. Configured EID-to-RLOC mappings	5
4.2. EID-to-RLOC Cache	6
4.3. Attacks using the data-plane	6
4.3.1. Attacks not leveraging on the LISP header	6
4.3.2. Attacks leveraging on the LISP header	8
4.4. Attacks using the control-plane	11
4.4.1. Attacks with Map-Request messages	11
4.4.2. Attacks with Map-Reply messages	12
4.4.3. Attacks with Map-Register messages	13
4.4.4. Attacks with Map-Notify messages	14
5. Attack categories	14
5.1. Intrusion	14
5.1.1. Description	14
5.1.2. Vectors	14
5.2. Denial of Service (DoS)	14
5.2.1. Description	14
5.2.2. Vectors	15
5.3. Subversion	15
5.3.1. Description	15
5.3.2. Vectors	15
6. Note on privacy	16
7. IANA Considerations	16
8. Security Considerations	16
9. Acknowledgments	17
10. References	17
10.1. Normative References	17
10.2. Informative References	18
Appendix A. Document Change Log	19
Authors' Addresses	21

1. Introduction

The Locator/ID Separation Protocol (LISP) is defined in [RFC6830]. The present document assesses the security level and identifies security threats in the LISP specification if LISP is deployed in the Internet (i.e., a public non-trustable environment). As a result of the performed analysis, the document discusses the severity of the threats and proposes recommendations to reach the same level of security in LISP than in Internet today (e.g., without LISP).

The document is composed of three main parts: the first discussing the LISP data-plane; while the second discussing the LISP control-

plane. The final part summarizes the recommendations to prevent the identified threats.

The LISP data-plane consists of LISP packet encapsulation, decapsulation, and forwarding and includes the map cache data structures used to perform these operations.

The LISP control-plane consists in the mapping distribution system, which can be one of the mapping distribution systems proposed so far (e.g., [RFC6830], [I-D.ietf-lisp-ddt], [RFC6836], [RFC6833], [I-D.meyer-lisp-cons], and [RFC6837]), and the Map-Request, Map-Reply, Map-Register, and Map-Notification messages.

This document does not consider all the possible uses of LISP as discussed in [RFC6830]. The document focuses on LISP unicast, including as well LISP Interworking [RFC6832], LISP-MS [RFC6833], and LISP Map-Versioning [RFC6834]. The reading of these documents is a prerequisite for understanding the present document.

Unless otherwise stated, the document assumes a generic IP service and does not discuss the difference, from a security viewpoint, between using IPv4 or IPv6.

2. On-path Attackers

On-path attackers are attackers that are able to capture and modify all the packets exchanged between an Ingress Tunnel Router (ITR) and an Egress Tunnel Router (ETR). To cope with such an attacker, cryptographic techniques such as those used by IPSec ([RFC4301]) are required. As with IP, LISP relies on higher layer cryptography to secure packet payloads from on path attacks, so this document does not consider on-path attackers in this document.

Similarly, a time-shifted attack is an attack where the attacker is temporarily on the path between two communicating hosts. While it is on-path, the attacker sends specially crafted packets or modifies packets exchanged by the communicating hosts in order to disturb the packet flow (e.g., by performing a man in the middle attack). An important issue for time-shifted attacks is the duration of the attack once the attacker has left the path between the two communicating hosts. We do not consider time-shifted attacks in this document.

3. Off-Path Attackers: Reference Environment

The reference environment shown in the figure below is considered throughout this document. There are two hosts attached to LISP routers: HA and HB. HA is attached to the two LISP xTRs LR1 and LR2, which in turn are attached to two different ISPs. HB is attached to the two LISP xTRs LR3 and LR4. HA and HB are the EIDs of the two hosts. LR1, LR2, LR3, and LR4 are the RLOCs of the xTRs. PxTR is a proxy xTR and MR/MS plays the roles of Map Server and/or Map Resolver.

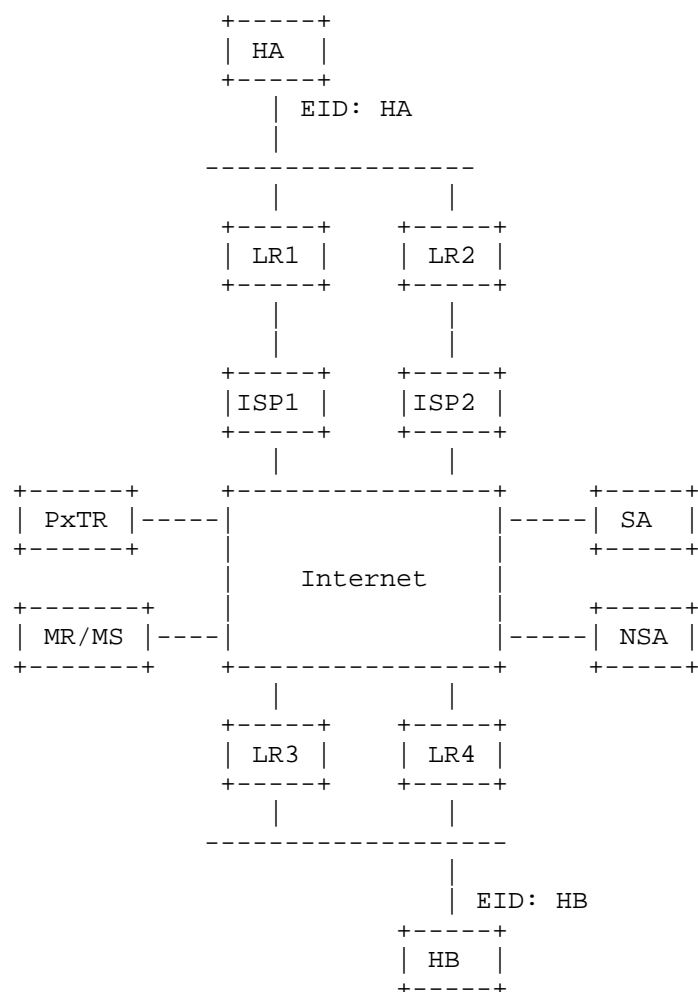


Figure 1: Reference Network

We consider two off-path attackers with different capabilities:

SA is an off-path attacker that is able to send spoofed packets, i.e., packets with a different source IP address than its assigned IP address. SA stands for Spoofing Attacker.

NSA is an off-path attacker that is only able to send packets whose source address is its assigned IP address. NSA stands for Non Spoofing Attacker.

It should be noted that with LISP, packet spoofing is slightly different than in the current Internet. Generally the term "spoofed packet" indicates a packet containing a source IP address that is not the one of the actual originator of the packet. Since LISP uses encapsulation, the spoofed address could be in the outer header as well as in the inner header, this translates in two types of spoofing:

EID Spoofing: the originator of the packet puts in it a spoofed EID. The packet will be normally encapsulated by the ITR of the site (or a PITR if the source site is not LISP enabled).

RLOC Spoofing: the originator of the packet generates directly a LISP-encapsulated packet with a spoofed source RLOC.

Note that the two types of spoofing are not mutually exclusive, rather all combinations are possible and could be used to perform different kind of attacks.

In the reference environment, both SA and NSA attackers are capable of sending LISP encapsulated data packets and LISP control packets. This means that SA is able to perform both RLOC and EID spoofing while NSA can only perform EID spoofing. They may also send other types of IP packets such as ICMP messages. We assume that both attackers can query the LISP mapping system (e.g., through a public Map Resolver) to obtain the mappings for both HA and HB.

4. Attack vectors

This section presents techniques that can be used by attackers to succeed attacks leveraging the LISP protocol and architecture. This section focuses on the techniques while Section 5 presents the attacks that can be succeeded while using these techniques.

4.1. Configured EID-to-RLOC mappings

Each xTR maintains a set of configured mappings related to the EID-Prefixes that are "behind" the xTR [RFC6830]. Where "behind" means

that at least one of the xTR's globally visible IP addresses is a RLOC for those EID-Prefixes.

As these mappings are determined by configuration. This means that the only way to attack this data structure is by gaining privileged access to the xTR. As such, it is out of the scope of LISP to propose any mechanism to protect routers and, hence, it is no further analyzed in this document.

4.2. EID-to-RLOC Cache

The EID-to-RLOC Cache (also called the Map-Cache) is the data structure that stores a copy of the mappings retrieved from a remote ETR's mapping via the LISP control-plane. Attacks against this data structure could happen either when the mappings are first installed in the cache or by corrupting (poisoning) the mappings already present in the cache.

This document calls "cache poisoning attack", any attack that alters the EID-to-RLOC Cache. Cache poisoning attacks are used to alter (any combination of) the following parts of mapping installed in the EID-to-RLOC Cache:

- o EID prefix
- o RLOC list
- o RLOC priority
- o RLOC weight
- o RLOC reachability
- o Mapping TTL
- o Mapping version
- o Mapping Instance ID

4.3. Attacks using the data-plane

The data-plane is constituted of the operations of encapsulation, decapsulation, and forwarding as well as the content of the EID-to-RLOC Cache and configured EID-to-RLOC mappings as specified in the original LISP document ([RFC6830]).

4.3.1. Attacks not leveraging on the LISP header

An attacker can inject packets into flows without using the LISP header, i.e., with the N, L, E, V, and I bits ([RFC6830]).

Taking notation of the reference environment notation (Figure 1), to inject a packet in the HA->HB flow, a spoofing off-path attacker (SA) could send a LISP encapsulated packet whose source is set to LR1 or LR2 and destination LR3 or LR4. The packet will reach HB as if the packet was sent by host HA. This is not different from today's Internet where a spoofing off-path attacker may inject data packets in any flow. A non-spoofing off-path attacker (NSA) could only send a packet whose source address is set to its assigned IP address. The destination address of the encapsulated packet could be LR3 or LR4.

4.3.1.1. Gleaning Attacks

In order to reduce the time required to obtain a mapping, [RFC6830] proposes the gleaning mechanism that allows an ITR to learn a mapping from the LISP data encapsulated packets and the Map-Request packets that it receives. LISP data encapsulated packet contains a source RLOC, destination RLOC, source EID and destination EID. When an ITR receives a data encapsulated packet coming from a source EID for which it does not already know a mapping, it may insert the mapping between the source RLOC and the source EID in its EID-to-RLOC Cache. Gleaning could also be used when an ITR receives a Map-Request as the Map-Request also contains a source EID address and a source RLOC. Once a gleaned entry has been added to the EID-to-RLOC cache, the LISP ITR sends a Map-Request to retrieve the mapping for the gleaned EID from the mapping system. [RFC6830] recommends storing the gleaned entries for only a few seconds.

An attacker can send LISP encapsulated packets to host HB with host HA's EID and if the xTRs that serve host HB do not store a mapping for host HA at that time. The xTR will store the gleaned entry and use it to return the packets sent by host HB. In parallel, the ETR will send a Map-Request to retrieve the mapping for HA but until the reception of the Map-Reply, host HB will exchange packets with the attacker instead of HA.

Similarly, if an off-path attacker knows that hosts HA and HB that resides in different sites will exchange information at time t the attacker could send to LR1 (resp. LR3) a LISP data encapsulated packet whose source RLOC is its IP address and contains an IP packet whose source is set to HB (resp. HA). The attacker chooses a packet that will not trigger an answer, for example the last part of a fragmented packet. Upon reception of these packets, LR1 and LR3 install gleaned entries that point to the attacker. If host HA is willing to establishes a flow with host HB at that time, the packets that they exchange will pass through the attacker as long as the gleaned entry is active on the xTRs.

By itself, an attack made solely using gleaning cannot last long, however it should be noted that with current network capacities, a large amount of packets might be exchanged during even a small fraction of time.

4.3.1.2. Threats concerning Interworking

[RFC6832] defines Proxy-ITR And Proxy-ETR network elements to allow LISP and non-LISP sites to communicate. The Proxy-ITR has functionality similar to the ITR, however, its main purpose is to encapsulate packets arriving from the DFZ in order to reach LISP sites. A Proxy-ETR has functionality similar to the ETR, however, its main purpose is to inject de-encapsulated packets in the DFZ in order to reach non-LISP Sites from LISP sites. As a PITR (resp. PETR) is a particular case of ITR (resp. ETR), it is subject to same attacks than ITRs (resp. ETR).

PxTRs can be targeted by attacks aiming to influence traffic between LISP and non-LISP sites but also to launch relay attacks.

It is worth to notice that when PITR and PETR functions are separated, attacks targeting nodes that collocate PITR and PETR functionality are ineffective.

4.3.2. Attacks leveraging on the LISP header

The main LISP document [RFC6830] defines several flags that modify the interpretation of the LISP header in data packets. In this section, we discuss how an off-path attacker could exploit this LISP header.

4.3.2.1. Attacks using the Locator Status Bits

When the L bit is set to 1, it indicates that the second 32-bits longword of the LISP header contains the Locator Status Bits. In this field, each bit position reflects the status of one of the RLOCs mapped to the source EID found in the encapsulated packet. In particular, a packet with the L bit set and all Locator Status Bits set to zero indicates that none of the locators of the encapsulated source EID are reachable. The reaction of a LISP ETR that receives such a packet is not clearly described in [RFC6830].

An attacker can send a data packet with the L bit set to 1 and some or all Locator Status Bits set to zero. Therefore, by blindly trusting the Locator Status Bits communication going on can be altered or forced to go through a particular set of locators.

4.3.2.2. Attacks using the Map-Version bit

The optional Map-Version bit is used to indicate whether the low-order 24 bits of the first 32 bits longword of the LISP header contain a Source and Destination Map-Version. When a LISP ETR receives a LISP encapsulated packet with the Map-Version bit set to 1, the following actions are taken:

- o It compares the Destination Map-Version found in the header with the current version of its own configured EID-to-RLOC mapping, for the destination EID found in the encapsulated packet. If the received Destination Map-Version is smaller (i.e., older) than the current version, the ETR should apply the SMR procedure described in [RFC6830] and send a Map-Request with the SMR bit set.
- o If a mapping exists in the EID-to-RLOC Cache for the source EID, then it compares the Map-Version of that entry with the Source Map-Version found in the header of the packet. If the stored mapping is older (i.e., the Map-Version is smaller) than the source version of the LISP encapsulated packet, the xTR should send a Map-Request for the source EID.

An off-path attacker could use the Map-Version bit to force an ETR to send Map-Request messages. The attacker could retrieve the current source and destination Map-Version for both HA and HB. Based on this information, it could send a spoofed packet with an older Source Map-Version or Destination Map-Version. If the size of the Map-Request message is larger than the size of the smallest LISP-encapsulated packet that could trigger such a message, this could lead to amplification attacks (see Section 4.4.1) so that more bandwidth is consumed on the target than the bandwidth necessary at the attacker side.

4.3.2.3. Attacks using the Nonce-Present and the Echo-Nonce bits

The Nonce-Present and Echo-Nonce bits are used when verifying the reachability of a remote ETR. Assume that LR3 wants to verify that LR1 receives the packets that it sends. LR3 can set the Echo-Nonce and the Nonce-Present bits in LISP data encapsulated packets and include a random nonce in these packets. Upon reception of these packets, LR1 will store the nonce sent by LR3 and echo it when it returns LISP encapsulated data packets to LR3.

A spoofing off-path attacker (SA) could interfere with this reachability test by sending two different types of packets:

1. LISP data encapsulated packets with the Nonce-Present bit set and a random nonce and the appropriate source and destination RLOCs.
2. LISP data encapsulated packets with the Nonce-Present and the Echo-Nonce bits both set and the appropriate source and destination RLOCs. These packets will force the receiving ETR to store the received nonce and echo it in the LISP encapsulated packets that it sends.

The first type of packet should not cause any major problem to ITRs. As the reachability test uses a 24 bits nonce, it is unlikely that an off-path attacker could send a single packet that causes an ITR to believe that the ETR it is testing is reachable while in reality it is not reachable. To increase the success likelihood of such attack, the attacker should create a massive amount of packets carrying all possible nonce values.

The second type of packet could be exploited to attack the nonce-based reachability test. Consider a spoofing off-path attacker (SA) that sends a continuous flow of spoofed LISP data encapsulated packets that contain the Nonce-Present and the Echo-Nonce bit and each packet contains a different random nonce. The ETR that receives such packets will continuously change the nonce that it returns to the remote ITR. If the remote ITR starts a nonce-reachability test, this test may fail because the ETR has received a spoofed LISP data encapsulated packet with a different random nonce and never echoes the real nonce. In this case the ITR will consider the ETR not reachable. The success of this test depends on the ratio between the amount of packets sent by the legitimate ITR and the spoofing off-path attacker (SA).

4.3.2.4. Attacks using the Instance ID bits

LISP allows to carry in its header a 24-bits value called "Instance ID" and used on the ITR to indicate which local Instance ID has been

used for encapsulation, while on the ETR can be used to select the forwarding table used for forwarding the decapsulated packet.

The Instance ID increases exposure to attacks ([RFC6169]) as if an off-path attacker can randomly guess a valid Instance ID value to get access to network that might not been accessible in normal conditions. However, the impact of such attack is directly on end-systems which is out of the scope of this document.

4.4. Attacks using the control-plane

In this section, we discuss the different types of attacks that could occur when an off-path attacker sends control-plane packets. We focus on the packets that are sent directly to the ETR and do not analyze the particularities of the different LISP indexing sub-system.

4.4.1. Attacks with Map-Request messages

An off-path attacker could send Map-Request packets to a victim ETR. In theory, a Map-Request packet is only used to solicit an answer and as such it should not lead to security problems. However, the LISP specification [RFC6830] contains several particularities that could be exploited by an off-path attacker.

The first possible exploitation is the RLOC record P bit. The RLOC record P bit is used to probe the reachability of remote ETRs. In our reference environment, LR3 could probe the reachability of LR1 by sending a Map-Request with the RLOC record P bit set. LR1 would reply by sending a Map-Reply message with the RLOC record P bit set and the same nonce as in the Map-Request message.

A spoofing off-path attacker (SA) could use the RLOC record P bit to force a victim ETR to send a Map-Reply to the spoofed source address of the Map-Request message. As the Map-Reply can be larger than the Map-Request message, there is a risk of amplification attack. Considering only IPv6 addresses, a Map-Request can be as small as 40 bytes, considering one single ITR address and no Mapping Protocol Data. The Map-Reply instead has a proportional to the maximum number of RLOCs in a mapping and maximum number of records in a Map-Reply. Since up to 255 RLOCs can be associated to an EID-Prefix and 255 records can be stored in a Map-Reply, the maximum size of a Map-Reply is thus above 1 MB, largely bigger than the message sent by the attacker. These numbers are however theoretical values not considering transport layer limitations and it is more likely that the reply will contain only one record with at most a dozen of locators, limiting so the amplification factor.

Similarly, if a non-spoofing off-path attacker (NSA) sends a Map-Request with the RLOC record P bit set, it will receive a Map-Reply with the RLOC record P bit set.

An amplification attack could be launched by a spoofing off-path attacker (SA) as follows. Consider an attacker SA and EID-Prefix 192.0.2.0/24 and a victim ITR, SA could send spoofed Map-Request messages whose source EID addresses are all the addresses inside 192.0.2.0/24 and source RLOC address is the victim ITR. Upon reception of these Map-Request messages, the ETR would send large Map-Reply messages for each of the addresses inside p/p back to the victim ITR.

The Map-Request message may also contain the SMR bit. Upon reception of a Map-Request message with the SMR bit, an ETR must return to the source of the Map-Request message a Map-Request message to retrieve the corresponding mapping. This raises similar problems as the RLOC record P bit discussed above except that as the Map-Request messages are smaller than Map-Reply messages, the risk of amplification attacks is reduced. This is not true anymore if the ETR append to the Map-Request messages its own Map-Records. This mechanism is meant to reduce the delay in mapping distribution since mapping information is provided in the Map-Request message.

Furthermore, appending Map-Records to Map-Request messages allows an off-path attacker to generate a (spoofed or not) Map-Request message and include in the Map-Reply portion of the message mapping for EID prefixes that it does not serve.

Moreover, attackers can use Map Resolver and/or Map Server network elements to perform relay attacks. Indeed, on the one hand, a Map Resolver is used to dispatch Map-Request to the mapping system and, on the other hand, a Map Server is used to dispatch Map-Requests coming from the mapping system to ETRs that are authoritative for the EID in the Map-Request.

4.4.2. Attacks with Map-Reply messages

In this section we analyze the attacks that could occur when an off-path attacker sends directly Map-Reply messages to ETRs without using one of the proposed LISP mapping systems.

There are two different types of Map-Reply messages:

Positive Map-Reply: These messages contain a Map-Record binding an EID-Prefix to one or more RLOCs.

Negative Map-Reply: These messages contain a Map-Record for an EID-Prefix with an empty locator-set and specifying an action, which may be either Drop, natively forward, or Send Map-Request.

Positive Map-Reply messages are used to map EID-Prefixes onto RLOCs. Negative Map-Reply messages are used to indicate non-LISP prefixes. ITRs can, if needed, be configured to send all traffic destined for non-LISP prefixes to a Proxy-ETR.

Most of the security of the Map-Reply messages depends on the 64 bits nonce that is included in a Map-Request and returned in the Map-Reply. If an ETR does not accept Map-Reply messages with an invalid nonce, the risk of attack is acceptable given the size of the nonce (64 bits). However, the nonce only confirms that the Map-Reply received was sent in response to a Map-Request sent, it does not validate the contents of that Map-Reply.

In addition, an attacker could perform EID-to-RLOC Cache overflow attack by de-aggregating (i.e., splitting an EID prefix into artificially smaller EID prefixes) either positive or negative mappings.

In presence of malicious ETRs, overclaiming attacks are possible. Such an attack happens when an ETR replies to a legitimate Map-Request message it received with a Map-Reply message that contains an EID-Prefix that is larger than the prefix owned by the site that encompasses the EID of the Map-Request. For instance if the prefix owned by the site is 192.0.2.0/25 but the Map-Reply contains a mapping for 192.0.2.0/24, then the mapping will influence packets destined to other EIDs than the one the LISP site has authority on.

A malicious ETR might also fragment its configured EID-to-RLOC mappings so that ITR's might have to install much more mappings than really necessary. This attack is called de-aggregation attack.

4.4.3. Attacks with Map-Register messages

Map-Register messages are sent by ETRs to indicate to the mapping system the EID prefixes associated to them. The Map-Register message provides an EID prefix and the list of ETRs that are able to provide Map-Replies for the EID covered by the EID prefix.

As Map-Register messages are protected by an authentication mechanism, only a compromised ETR can register itself to its allocated Map Server.

A compromised ETR can perform an overclaiming attack in order to influence the route followed by Map-Requests for EIDs outside the scope of its legitimate EID prefix.

A compromised ETR can also perform a deaggregation attack in order to register more EID prefixes than necessary to its Map Servers.

Similarly, a compromised Map Server can accept invalid registration or advertise invalid EID prefix to the indexing sub-system.

4.4.4. Attacks with Map-Notify messages

Map-Notify messages are sent by a Map Server to an ETR to acknowledge the good reception and processing of a Map-Register message.

An compromised ETR using EID that it is not authoritative for can send a Map-Register with the M-bit set and a spoofed source address to force the Map Server to send a Map-Notify message to the spoofed address and then succeed a relay attack. Similarly to the pair Map-Request/Map-Reply, the pair Map-Register/Map-Notify is protected by a nonce making it hard for an attacker to inject a falsified notification to an ETR to make this ETR believe that the registration succeeded while it has not.

5. Attack categories

5.1. Intrusion

5.1.1. Description

With an intrusion attack an attacker gains remote access to some resources (e.g., a host, a router, or a network) that are normally denied to her.

5.1.2. Vectors

Intrusion attacks can be mounted using:

- o Spoofing EID or RLOCs
- o Instance ID bits

5.2. Denial of Service (DoS)

5.2.1. Description

A Denial of Service (DoS) attack aims at disrupting a specific targeted service either by exhausting the resources of the victim up

to the point that it is not able to provide a reliable service to legit traffic and/or systems or by exploiting vulnerabilities to make the targeted service unable to operate properly.

5.2.2. Vectors

Denial of Service attacks can be mounted using

- o Gleaning
- o Interworking
- o Locator Status Bits
- o Map-Version bit
- o Nonce-Present and Echo-Nonce bits
- o Map-Request message
- o Map-Reply message
- o Map-Register message
- o Map-Notify message

5.3. Subversion

5.3.1. Description

With subversion an attacker can gain access (e.g., using eavesdropping or impersonation) to restricted or sensitive information such as passwords, session tokens, or any other confidential information. This type of attack is usually carried out in a way such that the target does not even notice the attack. When the attacker is positioned on the path of the target traffic, it is called a Man-in-the-Middle attack. However, this is not a requirement to carry out an eavesdropping attack. Indeed the attacker might be able, for instance through an intrusion attack on a weaker system, either to duplicate or even re-direct the traffic, in both cases having access to the raw packets.

5.3.2. Vectors

Subversion attacks can be mounted using

- o Gleaning

- o Locator Status Bits
- o Nonce-Present and the Echo-Nonce bits
- o Map-Request messages
- o Map-Reply messages

6. Note on privacy

As presented by [RFC6973], universal privacy considerations are impossible to establish as the privacy definition may vary from one to another. As a consequence, this document does not aim at identifying privacy issues related to the LISP protocol but it is necessary to highlight that security threats identified in this document could play a role in privacy threats as defined in section 5 of [RFC6973].

7. IANA Considerations

This document makes no request to IANA.

8. Security Considerations

This document is devoted to threat analysis of the Locator/Identifier Separation Protocol and is then a piece of choice to understand the security risks at stake while deploying LISP in non-trustable environment.

The purpose of this document is not to provide recommendations to protect against attacks, however most of threats can be prevented with careful deployment and configuration (e.g., filter) and also by applying the general rules in security that consist in activating only features that are necessary in the deployment and verifying the validity of the information obtained from third parties. More detailed recommendation are given in [book_chapter].

The control-plane is probably the most critical part of LISP from a security viewpoint and it is worth to notice that the specifications already offer authentication mechanism for Map-Register messages ([RFC6833]) and that [I-D.ietf-lisp-sec] and [I-D.ietf-lisp-ddt] are clearly going in the direction of a secure control-plane.

9. Acknowledgments

This document builds upon the draft of Marcelo Bagnulo ([I-D.bagnulo-lisp-threat]), where the flooding attack and the reference environment were first described.

The authors would like to thank Ronald Bonica, Albert Cabellos, Noel Chiappa, Florin Coras, Vina Ermagan, Dino Farinacci, Stephen Farrell, Joel Halpern, Emily Hiltzik, Darrel Lewis, Edward Lopez, Fabio Maino, Terry Manderson, and Jeff Wheeler for their comments.

This work has been partially supported by the INFSO-ICT-216372 TRILOGY Project (www.trilogy-project.org).

10. References

10.1. Normative References

- [RFC6169] Krishnan, S., Thaler, D., and J. Hoagland, "Security Concerns with IP Tunneling", RFC 6169, April 2011.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, January 2013.
- [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, July 2013.

10.2. Informative References

- [Chu] Jerry Chu, H., "Tuning TCP Parameters for the 21st Century", 75th IETF, Stockholm, July 2009, <<http://tools.ietf.org/wg/savi/>>.
- [I-D.bagnulo-lisp-threat] Bagnulo, M., "Preliminary LISP Threat Analysis", draft-bagnulo-lisp-threat-01 (work in progress), July 2007.
- [I-D.ietf-lisp-ddt] Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-01 (work in progress), March 2013.
- [I-D.ietf-lisp-sec] Maino, F., Ermagan, V., Cabellos-Aparicio, A., Saucez, D., and O. Bonaventure, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-04 (work in progress), October 2012.
- [I-D.ietf-tcpm-tcp-security] Gont, F., "Survey of Security Hardening Methods for Transmission Control Protocol (TCP) Implementations", draft-ietf-tcpm-tcp-security-03 (work in progress), March 2012.
- [I-D.meyer-lisp-cons] Brim, S., "LISP-CONS: A Content distribution Overlay Network Service for LISP", draft-meyer-lisp-cons-04 (work in progress), April 2008.
- [I-D.saucez-lisp-mapping-security] Saucez, D. and O. Bonaventure, "Securing LISP Mapping replies", draft-saucez-lisp-mapping-security-00 (work in progress), February 2011.
- [RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.
- [RFC5386] Williams, N. and M. Richardson, "Better-Than-Nothing Security: An Unauthenticated Mode of IPsec", RFC 5386, November 2008.
- [RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, February 2012.

- [SAVI] IETF, "Source Address Validation Improvements Working Group ", 2013, <<http://tools.ietf.org/wg/savi/>>.
- [Saucez09] Saucez, D. and L. Iannone, "How to mitigate the effect of scans on mapping systems ", Trilogy Summer School on Future Internet, 2009.
- [book_chapter] Saucez, D., Iannone, L., and O. Bonaventure, "The Map-and-Encap Locator/Identifier separation paradigm: a Security Analysis ", Solutions for Sustaining Scalability in Internet Growth, IGI Global, 2013.

Appendix A. Document Change Log

- o Version 08 Posted October 2013.
 - * Addition of a privacy consideration note.
 - * Editorial changes
- o Version 07 Posted October 2013.
 - * This version is updated according to the thorough review made during October 2013 LISP WG interim meeting.
 - * Brief recommendations put in the security consideration section.
 - * Editorial changes
- o Version 06 Posted October 2013.
 - * Complete restructuration, temporary version to be used at October 2013 interim meeting.
- o Version 05 Posted August 2013.
 - * Removal of severity levels to become a short recommendation to reduce the risk of the discussed threat.
- o Version 04 Posted February 2013.
 - * Clear statement that the document compares threats of public LISP deployments with threats in the current Internet architecture.

- * Addition of a severity level discussion at the end of each section.
- * Addressed comments from V. Ermagan and D. Lewis' reviews.
- * Updated References.
- * Further editorial polishing.
- o Version 03 Posted October 2012.
 - * Dropped Reference to RFC 2119 notation because it is not actually used in the document.
 - * Deleted future plans section.
 - * Updated References
 - * Deleted/Modified sentences referring to the early status of the LISP WG and documents at the time of writing early versions of the document.
 - * Further editorial polishing.
 - * Fixed all ID nits.
- o Version 02 Posted September 2012.
 - * Added a new attack that combines overclaiming and de-aggregation (see Section 4.4.2).
 - * Editorial polishing.
- o Version 01 Posted February 2012.
 - * Added discussion on LISP-DDT.
- o Version 00 Posted July 2011.
 - * Added discussion on LISP-MS>.
 - * Added discussion on Instance ID in Section 4.3.2.
 - * Editorial polishing of the whole document.
 - * Added "Change Log" appendix to keep track of main changes.
 - * Renamed "draft-saucez-lisp-security-03.txt".

Authors' Addresses

Damien Saucez
INRIA
2004 route des Lucioles BP 93
06902 Sophia Antipolis Cedex
France

Email: damien.saucez@inria.fr

Luigi Iannone
Telecom ParisTech
23, Avenue d'Italie, CS 51327
75214 PARIS Cedex 13
France

Email: luigi.iannone@telecom-paristech.fr

Olivier Bonaventure
Universite catholique de Louvain
Place St. Barbe 2
Louvain la Neuve
Belgium

Email: olivier.bonaventure@uclouvain.be

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 1, 2016

D. Saucez
INRIA
L. Iannone
Telecom ParisTech
O. Bonaventure
Universite catholique de Louvain
January 29, 2016

LISP Threats Analysis
draft-ietf-lisp-threats-15.txt

Abstract

This document provides a threat analysis of the Locator/Identifier Separation Protocol (LISP).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 1, 2016.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Threat model	3
2.1. Attacker's Operation Modes	4
2.1.1. On-path vs. Off-path Attackers	4
2.1.2. Internal vs. External Attackers	4
2.1.3. Live vs. Time-shifted attackers	5
2.1.4. Control-plane vs. Data-plane attackers	5
2.1.5. Cross mode attackers	5
2.2. Threat categories	5
2.2.1. Replay attack	5
2.2.2. Packet manipulation	6
2.2.3. Packet interception and suppression	6
2.2.4. Spoofing	6
2.2.5. Rogue attack	7
2.2.6. Denial of Service (DoS) attack	7
2.2.7. Performance attack	7
2.2.8. Intrusion attack	7
2.2.9. Amplification attack	7
2.2.10. Passive Monitoring Attacks	8
2.2.11. Multi-category attacks	8
3. Attack vectors	8
3.1. Gleaning	8
3.2. Locator Status Bits	9
3.3. Map-Version	10
3.4. Routing Locator Reachability	11
3.5. Instance ID	12
3.6. Interworking	12
3.7. Map-Request messages	12
3.8. Map-Reply messages	13
3.9. Map-Register messages	15
3.10. Map-Notify messages	15
4. Note on Privacy	15
5. Threats Mitigation	16
6. Security Considerations	17
7. IANA Considerations	17
8. Acknowledgments	17
9. References	17
9.1. Normative References	17
9.2. Informative References	18
Appendix A. Document Change Log (to be removed on publication) .	19
Authors' Addresses	21

1. Introduction

The Locator/ID Separation Protocol (LISP) is specified in [RFC6830]. This document provides an assessment of the potential security threats for the current LISP specifications if LISP is deployed in the Internet (i.e., a public non-trustable environment).

The document is composed of three main parts: the first defines a general threat model that attackers use to mount attacks. The second part, using this threat model, describes the techniques based on the LISP protocol and LISP architecture that attackers may use to construct attacks. The third part discusses mitigation techniques and general solutions to protect the LISP protocol and architecture from attacks.

This document does not consider all the possible uses of LISP as discussed in [RFC6830] and [RFC7215] and does not cover threats due to specific implementations. The document focuses on LISP unicast, including as well LISP Interworking [RFC6832], LISP Map-Server [RFC6833], and LISP Map-Versioning [RFC6834]. Additional threats may be discovered in the future while deployment continues. The reader is assumed to be familiar with these documents for understanding the present document.

This document assumes a generic IP service and does not discuss the difference, from a security viewpoint, between using IPv4 or IPv6.

2. Threat model

This document assumes that attackers can be located anywhere in the Internet (either in LISP sites or outside LISP sites) and that attacks can be mounted either by a single attacker or by the collusion of several attackers.

An attacker is a malicious entity that performs the action of attacking a target in a network where LISP is (partially) deployed by leveraging the LISP protocol and/or architecture.

An attack is the action of performing an illegitimate action on a target in a network where LISP is (partially) deployed.

The target of an attack is the entity (i.e., a device connected to the network or a network) that is aimed to undergo the consequences of an attack. Other entities can potentially undergo side effects of an attack, even though they are not directly targeted by the attack. The target of an attack can be selected specifically, i.e., a particular entity, or arbitrarily, i.e., any entity. Finally, an

attacker can aim at attacking one or several targets with a single attack.

Section 2.1 specifies the different modes of operation that attackers can follow to mount attacks and Section 2.2 specifies the different categories of attacks that attackers can build.

2.1. Attacker's Operation Modes

In this document attackers are classified according to their modes of operation, i.e., the temporal and spacial diversity of the attacker. These modes are not mutually exclusive, they can be used by attackers in any combination, and other modes may be discovered in the future. Further, attackers are not at all bound by our classification scheme, so implementers and those deploying will always need to do additional risk analysis for themselves.

2.1.1. On-path vs. Off-path Attackers

On-path attackers, also known as Men-in-the-Middle, are able to intercept and modify packets between legitimate communicating entities. On-path attackers are located either directly on the normal communication path (either by gaining access to a node on the path or by placing themselves directly on the path) or outside the location path but manage to deviate (or gain a copy of) packets sent between the communication entities. On-path attackers hence mount their attacks by modifying packets initially sent legitimately between communication entities.

An attacker is called off-path attacker if it does not have access to packets exchanged during the communication or if there is no communication. In order for their attacks to succeed, off-path attackers must hence generate packets and inject them in the network.

2.1.2. Internal vs. External Attackers

An internal attacker launches its attack from a node located within a legitimate LISP site. Such an attacker is either a legitimate node of the site or it exploits a vulnerability to gain access to a legitimate node in the site. Because of their location, internal attackers are trusted by the site they are in.

On the contrary, an external attacker launches its attacks from the outside of a legitimate LISP site.

2.1.3. Live vs. Time-shifted attackers

A live attacker mounts attacks for which it must remain connected as long as the attack is mounted. In other words, the attacker must remain active for the whole duration of the attack. Consequently, the attack ends as soon as the attacker (or the used attack vector) is neutralized.

On the contrary, a time-shifted attacker mounts attacks that remain active after it disconnects from the Internet.

2.1.4. Control-plane vs. Data-plane attackers

A control-plane attacker mounts its attack by using control-plane functionalities, typically the mapping system.

A data-plane attacker mounts its attack by using data-plane functionalities.

As there is no complete isolation between the control-plane and the data-plane, an attacker can operate in the control-plane (or data-plane) to mount attacks targeting the data-plane (or control-plane) or keep the attacked and targeted planes at the same layer (i.e., from control-plane to control-plane or from data-plane to data-plane).

2.1.5. Cross mode attackers

The attacker modes of operation are not mutually exclusive and hence attackers can combine them to mount attacks.

For example, an attacker can launch an attack using the control-plane directly from within a LISP site to which it is able to get temporary access (i.e., internal + control-plane attacker) to create a vulnerability on its target and later on (i.e., time-shifted + external attacker) mount an attack on the data plane (i.e., data-plane attacker) that leverages the vulnerability.

2.2. Threat categories

Attacks can be classified according to the nine following categories. These categories are not mutually exclusive and can be used by attackers in any combination.

2.2.1. Replay attack

A replay attack happens when an attacker retransmits at a later time, and without modifying it, a packet (or a sequence of packets) that

has already been transmitted.

2.2.2. Packet manipulation

A packet manipulation attack happens when an attacker receives a packet, modifies the packet (i.e., changes some information contained in the packet) and finally transmits the packet to its final destination that can be the initial destination of the packet or a different one.

2.2.3. Packet interception and suppression

In a packet interception and suppression attack, the attacker captures the packet and drops it before it can reach its final destination.

2.2.4. Spoofing

With a spoofing attack, the attacker injects packets in the network pretending to be another node. Spoofing attacks are made by forging source addresses in packets.

It should be noted that with LISP, packet spoofing is similar to spoofing with any other existing tunneling technology currently deployed in the Internet. Generally the term "spoofed packet" indicates a packet containing a source IP address that is not the actual originator of the packet. Hence, since LISP uses encapsulation, the spoofed address could be in the outer header as well as in the inner header, this translates to two types of spoofing.

Inner address spoofing: the attacker uses encapsulation and uses a spoofed source address in the inner packet. In case of data-plane LISP encapsulation, that corresponds to spoofing the source EID (End-point IDentifier) address of the encapsulated packet.

Outer address spoofing: the attacker does not use encapsulation and spoofs the source address of the packet. In case of data-plane LISP encapsulation, that corresponds to spoofing the source RLOC (Routing LOCator) address of the encapsulated packet.

Note that the two types of spoofing are not mutually exclusive, rather all combinations are possible and could be used to perform different kinds of attacks. For example, an attacker outside a LISP site can generate a packet with a forged source IP address (i.e., outer address spoofing) and forward it to a LISP destination. The packet is then eventually encapsulated by a PITR (Proxy Ingress

Tunnel Router) so that once encapsulated the attack corresponds to a inner address spoofing. One can also imagine an attacker forging a packet with encapsulation where both inner and outer source addresses are spoofed.

It is important to note that the combination of inner and outer spoofing makes the identification of the attacker complex as the packet may not contain information that allows to detect the origin of the attack.

2.2.5. Rogue attack

In a rogue attack the attacker manages to appear as a legitimate source of information, without faking its identity (as opposed to a spoofing attacker).

2.2.6. Denial of Service (DoS) attack

A Denial of Service (DoS) attack aims at disrupting a specific targeted service to make it unable to operate properly.

2.2.7. Performance attack

A performance attacks aims at exploiting computational resources (e.g., memory, processor) of a targeted node so as to make it unable to operate properly.

2.2.8. Intrusion attack

In an intrusion attack, the attacker gains remote access to a resource (e.g., a host, a router, or a network) or information that it legitimately should not have access. Intrusion attacks can lead to privacy leakages.

2.2.9. Amplification attack

In an amplification attack, the traffic generated by the target of the attack in response to the attack is larger than the traffic that the attacker must generate.

In some cases, the data-plane can be several orders of magnitude faster than the control-plane at processing packets. This difference can be exploited to overload the control-plane via the data-plane without overloading the data-plane.

2.2.10. Passive Monitoring Attacks

An attacker can use pervasive monitoring, which is a technical attack [RFC7258], targeting information about LISP traffic that may or not be used to mount other type of attacks.

2.2.11. Multi-category attacks

Attacks categories are not mutually exclusive and any combination can be used to perform specific attacks.

For example, one can mount a rogue attack to perform a performance attack starving the memory of an ITR (Ingress Tunnel Router) resulting in a DoS (Denial-of-Service) on the ITR.

3. Attack vectors

This section presents attack techniques that may be used by attackers when leveraging the LISP protocol and/or architecture.

3.1. Gleaning

To reduce the time required to obtain a mapping, the optional gleaning mechanism defined for LISP allows an xTR (Ingress and/or Egress Tunnel Router) to directly learn a mapping from the LISP data encapsulated packets and the Map-Request packets that it receives. LISP encapsulated data packets contain a source RLOC, destination RLOC, source EID and destination EID. When an xTR receives an encapsulated data packet coming from a source EID for which it does not already know a mapping, it may insert the mapping between the source RLOC and the source EID in its EID-to-RLOC Cache. The same technique can be used when an xTR receives a Map-Request as the Map-Request also contains a source EID address and a source RLOC. Once a gleaned entry has been added to the EID-to-RLOC cache, the xTR sends a Map-Request to retrieve the actual mapping for the gleaned EID from the mapping system.

If a packet injected by an off-path attacker and with a spoofed inner address is gleaned by an xTR then the attacker may divert the traffic meant to be delivered to the spoofed EID as long as the gleaned entry is used by the xTR. This attack can be used as part of replay, packet manipulation, packet interception and suppression, or DoS attacks as the packets are sent to the attacker.

If the packet sent by the attacker contains a spoofed outer address instead of a spoofed inner address then it can achieve a DoS or a performance attack as the traffic normally destined to the attacker

will be redirected to the spoofed source RLOC. Such traffic may overload the owner of the spoofed source RLOC, preventing it from operating properly.

If the packet injected uses both inner and outer spoofing, the attacker can achieve a spoofing, a performance, or an amplification attack as traffic normally destined to the spoofed EID address will be sent to the spoofed RLOC address. If the attacked LISP site also generates traffic to the spoofed EID address, such traffic may have a positive amplification factor.

A gleaning attack does not only impact the data-plane but can also have repercussions on the control-plane as a Map-Request is sent after the creation of a gleaned entry. The attacker can then achieve DoS and performance attacks on the control-plane. For example, if an attacker sends a packet for each address of a prefix not yet cached in the EID-to-RLOC cache of an xTR, the xTR will potentially send a Map-Request for each such packet until the mapping is installed which leads to an over-utilisation of the control-plane as each packet generates a control-plane event. In order for this attack to succeed, the attacker may not need to use spoofing. This issue can occur even if gleaning is turned off since whether or not gleaning is used as the ITR may need to send a Map-Request in response to incoming packets whose EID is not currently in the cache.

Gleaning attacks are fundamentally involving a time-shifted mode of operation as the attack may last as long as the gleaned entry is kept by the targeted xTR. RFC 6830 [RFC6830] recommends to store the gleaned entries for only a few seconds which limits the duration of the attack.

Gleaning attacks always involve external data-plane attackers but results in attacks on either the control-plane or data-plane.

Note, the outer spoofed address does not need to be the RLOC of a LISP site, it may be any address.

3.2. Locator Status Bits

When the L bit in the LISP header is set to 1, it indicates that the second 32-bits longword of the LISP header contains the Locator Status Bits. In this field, each bit position reflects the status of one of the RLOCs mapped to the source EID found in the encapsulated packet. The reaction of a LISP xTR that receives such a packet is left as operational choice in [RFC6830].

When an attacker sends a LISP encapsulated packet with an illegitimately crafted LSB to an xTR, it can influence the xTR's

choice of the locators for the prefix associated to the source EID. In case of an off-path attacker, the attacker must inject a forged packet in the network with a spoofed inner address. An on-path attacker can manipulate the LSB of legitimate packets passing through it and hence does not need to use spoofing. Instead of manipulating the LSB field, an on-path attacker can also obtain the same result of injecting packets with invalid LSB values by replaying packets.

The LSB field can be leveraged to mount a DoS attack by either declaring all RLOCs as unreachable (all LSB set to 0), or by concentrating all the traffic to one RLOC (e.g., all but one LSB set to 0) and hence overloading the RLOC concentrating all the traffic from the xTR, or by forcing packets to be sent to RLOCs that are actually not reachable (e.g., invert LSB values).

The LSB field can also be used to mount a replay, a packet manipulation, or a packet interception and suppression attack. Indeed, if the attacker manages to be on the path between the xTR and one of the RLOCs specified in the mapping, forcing packets to go via that RLOC implies that the attacker will gain access to the packets.

Attacks using the LSB are fundamentally involving a time-shifted mode of operation as the attack may last as long as the reachability information gathered from the LSB is used by the xTR to decide the RLOCs to be used.

3.3. Map-Version

When the Map-Version bit of the LISP header is set to 1, it indicates that the low-order 24 bits of the first 32 bits longword of the LISP header contain a Source and Destination Map-Version. When a LISP xTR receives a LISP encapsulated packet with the Map-Version bit set to 1, the following actions are taken:

- o It compares the Destination Map-Version found in the header with the current version of its own configured EID-to-RLOC mapping, for the destination EID found in the encapsulated packet. If the received Destination Map-Version is smaller (i.e., older) than the current version, the ETR should apply the SMR (Solicit-Map-Request) procedure described in [RFC6830] and send a Map-Request with the SMR bit set.
- o If a mapping exists in the EID-to-RLOC Cache for the source EID, then it compares the Map-Version of that entry with the Source Map-Version found in the header of the packet. If the stored mapping is older (i.e., the Map-Version is smaller) than the source version of the LISP encapsulated packet, the xTR should send a Map-Request for the source EID.

A cross-mode attacker can use the Map-Version bit to mount a DoS attack, an amplification attack, or a spoofing attack. For instance if the mapping cached at the xTR is outdated, the xTR will send a Map-Request to retrieve the new mapping which can yield to a DoS attack (by excess of signalling traffic) or an amplification attack if the data-plane packet sent by the attacker is smaller, or otherwise uses fewer resources, than the control-plane packets sent in response to the attacker's packet. With a spoofing attack, and if the xTR considers that the spoofed ITR has an outdated mapping, it will send an SMR to the spoofed ITR which can result in performance, amplification, or DoS attack as well.

Map-Version attackers are inherently cross mode as the Map-Version is a method to put control information in the data-plane. Moreover, this vector involves live attackers. Nevertheless, on-path attackers do not have specific advantage over off-path attackers.

3.4. Routing Locator Reachability

The Nonce-Present and Echo-Nonce bits in the LISP header are used to verify the reachability of an xTR. A testing xTR sets the Echo-Nonce and the Nonce-Present bits in LISP data encapsulated packets and include a random nonce in the LISP header of packets. Upon reception of these packets, the tested xTR stores the nonce and echoes it whenever it returns a LISP encapsulated data packets to the testing xTR. The reception of the echoed nonce confirms that the tested xTR is reachable.

An attacker can interfere with the reachability test by sending two different types of packets:

1. LISP data encapsulated packets with the Nonce-Present bit set and a random nonce. Such packets are normally used in response to a reachability test.
2. LISP data encapsulated packets with the Nonce-Present and the Echo-Nonce bits both set. These packets will force the receiving ETR to store the received nonce and echo it in the LISP encapsulated packets that it sends. These packets are normally used as a trigger for a reachability test.

The first type of packets are used to make xTRs think that an other xTR is reachable while it is not. It is hence a way to mount a DoS attack (i.e., the ITR will send its packet to a non-reachable ETR when it should use another one).

The second type of packets could be exploited to attack the nonce-based reachability test. If the attacker sends a continuous flow of

packets that each have a different random nonce, the ETR that receives such packets will continuously change the nonce that it returns to the remote ITR, which can yield to a performance attack. If the remote ITR tries a nonce-reachability test, this test may fail because the ETR may echo an invalid nonce. This hence yields to a DoS attack.

In the case of an on-path attacker, a packet manipulation attack is necessary to mount the attack. To mount such an attack, an off-path attacker must mount an outer address spoofing attack.

If an xTR chooses to periodically check with active probes the liveness of entries in its EID-to-RLOC cache (as described in section 6.3 of [RFC6830]), then this may amplify the attack that caused the insertion of entries being checked.

3.5. Instance ID

LISP allows to carry in its header a 24-bits value called Instance ID and used on the ITR to indicate which local Instance ID has been used for encapsulation, while on the ETR the instance ID decides the forwarding table to use to forward the decapsulated packet in the LISP site.

An attacker (either a control-plane or data-plane attacker) can use the instance ID functionality to mount an intrusion attack.

3.6. Interworking

[RFC6832] defines Proxy-ITR and Proxy-ETR network elements to allow LISP and non-LISP sites to communicate. The Proxy-ITR has functionality similar to the ITR, however, its main purpose is to encapsulate packets arriving from the DFZ (Default-Free Zone) in order to reach LISP sites. A PETR (Proxy Egress Tunnel Router) has functionality similar to the ETR, however, its main purpose is to inject de-encapsulated packets in the DFZ in order to reach non-LISP sites from LISP sites. As a PITR (or PETR) is a particular case of ITR (or ETR), it is subject to similar attacks as ITRs (or ETRs).

As any other system relying on proxies, LISP interworking can be used by attackers to hide their exact origin in the network.

3.7. Map-Request messages

A control-plane off-path attacker can exploit Map-Request messages to mount DoS, performance, or amplification attacks. By sending Map-Request messages at high rate, the attacker can overload nodes involved in the mapping system. For instance sending Map-Requests at

high rate can considerably increase the state maintained in a Map-Resolver or consume CPU cycles on ETRs that have to process the Map-Request packets they receive in their slow path (i.e., performance or DoS attack). When the Map-Reply packet is larger than the Map-Request sent by the attacker, that yields to an amplification attack. The attacker can combine the attack with a spoofing attack to overload the node to which the spoofed address is actually attached.

Note, if the attacker sets the P bit (Probe Bit) in the Map-Request, it will cause legitimately sending the Map-Request directly to the ETR instead of passing through the mapping system.

The SMR bit can be used to mount a variant of these attacks.

For efficiency reasons, Map-Records can be appended to Map-Request messages. When an xTR receives a Map-Request with appended Map-Records, it does the same operations as for the other Map-Request messages and so is subject to the same attacks. However, it also installs in its EID-to-RLOC cache the Map-Records contained in the Map-Request. An attacker can then use this vector to force the installation of mappings in its target xTR. Consequently, the EID-to-RLOC cache of the xTR is polluted by potentially forged mappings allowing the attacker to mount any of the attacks categorized in Section 2.2 (see Section 3.8 for more details). Note, the attacker does not need to forge the mappings present in the Map-Request to achieve a performance or DoS attack. Indeed, if the attacker owns a large enough EID prefix it can de-aggregate it in many small prefixes, each corresponding to another mapping and it installs them in the xTR cache by mean of the Map-Request.

Moreover, attackers can use Map Resolver and/or Map Server network elements to relay its attacks and hide the origin of the attack. Indeed, on the one hand, a Map Resolver is used to dispatch Map-Request to the mapping system and, on the other hand, a Map Server is used to dispatch Map-Requests coming from the mapping system to ETRs that are authoritative for the EID in the Map-Request.

3.8. Map-Reply messages

Most of the security risks associated with Map-Reply messages will depend on the 64 bits nonce that is included in a Map-Request and returned in the Map-Reply. Given the size of the nonce (64 bits), if best current practice is used [RFC4086] and if an ETR does not accept Map-Reply messages with an invalid nonce, the risk of an off-path attack is limited. Nevertheless, the nonce only confirms that the Map-Reply received was sent in response to a Map-Request sent, it does not validate the contents of that Map-Reply.

If an attacker manages to send a valid (i.e., in response to a Map-Request and with the correct nonce) Map-Reply to an ITR, then it can perform any of the attacks categorised in Section 2.2 as it can inject forged mappings directly in the ITR EID-to-RLOC cache. For instance, if the mapping injected to the ITR points to the address of a node controlled by the attacker, it can mount replay, packet manipulation, packet interception and suppression, or DoS attacks, as it will receive every packet destined to a destination lying in the EID prefix of the injected mapping. In addition, the attacker can inject a plethora of mappings in the ITR to mount a performance attack by filling up the EID-to-RLOC cache of the ITR. The attacker can also mount an amplification attack if the ITR at that time is sending a large number of packets to the EIDs matching the injected mapping. In this case, the RLOC address associated to the mapping is the address of the real target of the attacker and so all the traffic of the ITR will be sent to the target which means that with one single packet the attacker may generate very high traffic towards its final target.

If the attacker is a valid ETR in the system, it can mount a rogue attack if it uses prefixes over-claiming. In such a scenario, the attacker ETR replies to a legitimate Map-Request message which it received with a Map-Reply message that contains an EID-Prefix that is larger than the prefix owned by the attacker. For example if the owned prefix is 192.0.2.0/25 but the Map-Reply contains a mapping for 192.0.2.0/24, then the mapping will influence packets destined to other EIDs than the one attacker has authority on. With such technique, the attacker can mount the attacks presented above as it can (partially) control the mappings installed on its target ITR. To force its target ITR to send a Map-Request, nothing prevents the attacker to initiate some communication with the ITR. This method can be used by internal attackers that want to control the mappings installed in their site. To that aim, they simply have to collude with an external attacker ready to over-claim prefixes on behalf of the internal attacker.

Note, when the Map-Reply is in response to a Map-Request sent via the mapping system (i.e., not send directly from the ITR to an ETR), the attacker does not need to use a spoofing attack to achieve its attack as by design the source IP address of a Map-Reply is not known in advance by the ITR.

Map-Request and Map-Reply messages are exposed to any type of attackers, on-path or off-path but also external or internal attackers. Also, even though they are control message, they can be leveraged by data-plane attackers. As the decision of removing mappings is based on the TTL indicated in the mapping, time-shifted attackers can take advantage of injecting forged mappings as well.

3.9. Map-Register messages

Map-Register messages are sent by ETRs to Map Servers to indicate to the mapping system the EID prefixes associated to them. The Map-Register message provides an EID prefix and the list of ETRs that are able to provide Map-Replies for the EID covered by the EID prefix.

As Map-Register messages are protected by an authentication mechanism, only a compromised ETR can register itself to its allocated Map Server.

A compromised ETR can over-claim the prefix it owns in order to influence the route followed by Map-Requests for EIDs outside the scope of its legitimate EID prefix (see Section 3.8 for the list of over-claiming attacks).

A compromised ETR can also de-aggregate its EID prefix in order to register more EID prefixes than necessary to its Map Servers (see Section 3.7 for the impact of de-aggregation of prefixes by an attacker).

Similarly, a compromised Map Server can accept an invalid registration or advertise an invalid EID prefix to the mapping system.

3.10. Map-Notify messages

Map-Notify messages are sent by a Map Server to an ETR to acknowledge the reception and processing of a Map-Register message.

Similarly to the pair Map-Request/Map-Reply, the pair Map-Register/Map-Notify is protected by a nonce making it difficult for an attacker to inject a falsified notification to an ETR to make this ETR believe that the registration succeeded when it has not.

4. Note on Privacy

As reviewed in [RFC6973], universal privacy considerations are difficult to establish as the privacy definitions may vary for different scenarios. As a consequence, this document does not aim at identifying privacy issues related to the LISP protocol but the security threats identified in this document could play a role in privacy threats as defined in section 5 of [RFC6973].

Similar to public deployments of any other control plane protocols, in an Internet deployment, LISP mappings are public and hence provide information about the infrastructure and reachability of LISP sites

(i.e., the addresses of the edge routers). Depending upon deployment details, LISP map replies might or might not provide finer grained and more detailed information than is available with currently deployed routing and control protocols.

5. Threats Mitigation

Most of the above threats can be mitigated with careful deployment and configuration (e.g., filter) and also by applying the general rules of security, e.g. only activating features that are necessary for the deployment and verifying the validity of the information obtained from third parties.

The control-plane is the most critical part of LISP from a security viewpoint and it is worth to notice that the LISP specifications already offer an authentication mechanism for mappings registration ([RFC6833]). This mechanism, combined with LISP-SEC [I-D.ietf-lisp-sec], strongly mitigates threats in non-trustable environments such as the Internet. Moreover, an authentication data field for Map-Request messages and Encapsulated Control messages was allocated [RFC6830]. This field provides a general authentication mechanism technique for the LISP control-plane which future specifications may use while staying backward compatible. The exact technique still has to be designed and defined. To maximally mitigate the threats on the mapping system, authentication must be used, whenever possible, for both Map-Request and Map-Reply messages and for messages exchanged internally among elements of the mapping system, such as specified in [I-D.ietf-lisp-sec] and [I-D.ietf-lisp-ddt].

Systematically applying filters and rate-limitation, as proposed in [RFC6830], will mitigate most of the threats presented in this document. In order to minimise the risk of overloading the control-plane with actions triggered from data-plane events, such actions should be rate limited.

Moreover, all information opportunistically learned (e.g., with LSB or gleaning) should be used with care until they are verified. For example, a reachability change learned with LSB should not be used directly to decide the destination RLOC, but instead should trigger a rate-limited reachability test. Similarly, a gleaned entry should be used only for the flow that triggered the gleaning procedure until the gleaned entry has been verified [Trilogy].

6. Security Considerations

This document provides a threat analysis and proposes mitigation techniques for the Locator/Identifier Separation Protocol.

7. IANA Considerations

This document makes no request to IANA.

8. Acknowledgments

This document builds upon the document of Marcelo Bagnulo ([I-D.bagnulo-lisp-threat]), where the flooding attack and the reference environment was first described.

The authors would like to thank Deborah Brungard, Ronald Bonica, Albert Cabellos, Ross Callon, Noel Chiappa, Florin Coras, Vina Ermagan, Dino Farinacci, Stephen Farrell, Joel Halpern, Emily Hiltzik, Darrel Lewis, Edward Lopez, Fabio Maino, Terry Manderson, and Jeff Wheeler for their comments.

This work has been partially supported by the INFISO-ICT-216372 TRILOGY Project (www.trilogy-project.org).

The work of Luigi Iannone has been partially supported by the ANR-13-INFN-0009 LISP-Lab Project (www.lisp-lab.org) and the EIT KIC ICT-Labs SOFNETS Project.

9. References

9.1. Normative References

- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <<http://www.rfc-editor.org/info/rfc6830>>.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, DOI 10.17487/RFC6832, January 2013, <<http://www.rfc-editor.org/info/rfc6832>>.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833,

DOI 10.17487/RFC6833, January 2013,
<<http://www.rfc-editor.org/info/rfc6833>>.

[RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <<http://www.rfc-editor.org/info/rfc6834>>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <<http://www.rfc-editor.org/info/rfc6973>>.

9.2. Informative References

- [I-D.bagnulo-lisp-threat]
Bagnulo, M., "Preliminary LISP Threat Analysis", draft-bagnulo-lisp-threat-01 (work in progress), July 2007.
- [I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-03 (work in progress), April 2015.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-09 (work in progress), October 2015.
- [RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, DOI 10.17487/RFC4086, June 2005, <<http://www.rfc-editor.org/info/rfc4086>>.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", RFC 7215, DOI 10.17487/RFC7215, April 2014, <<http://www.rfc-editor.org/info/rfc7215>>.
- [RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <<http://www.rfc-editor.org/info/rfc7258>>.
- [Trilogy] Saucez, D. and L. Iannone, "How to mitigate the effect of scans on mapping systems", Trilogy Future Internet Summer

School., 2009.

Appendix A. Document Change Log (to be removed on publication)

- o Version 15 Posted January 2016.
 - * Few changes to address Stephen Farrel comments as part of the IESG Review.
- o Version 14 Posted December 2015.
 - * Editorial changes according to Deborah Brungard's (Routing AD) review.
- o Version 13 Posted August 2015.
 - * Keepalive version.
- o Version 12 Posted March 2015.
 - * Addressed comments by Ross Callon on the mailing list (<http://www.ietf.org/mail-archive/web/lisp/current/msg05829.html>).
 - * Addition of a section discussing mitigation techniques for deployments in non-trustable environments.
- o Version 11 Posted December 2014.
 - * Editorial polishing. Clarifications added in few points.
- o Version 10 Posted July 2014.
 - * Document completely remodelled according to the discussions on the mailing list in the thread <http://www.ietf.org/mail-archive/web/lisp/current/msg05206.html> and to address comments from Ronald Bonica and Ross Callon.
- o Version 09 Posted March 2014.
 - * Updated document according to the review of A. Cabellos.
- o Version 08 Posted October 2013.
 - * Addition of a privacy consideration note.
 - * Editorial changes

- o Version 07 Posted October 2013.
 - * This version is updated according to the thorough review made during October 2013 LISP WG interim meeting.
 - * Brief recommendations put in the security consideration section.
 - * Editorial changes
- o Version 06 Posted October 2013.
 - * Complete restructuration, temporary version to be used at October 2013 interim meeting.
- o Version 05 Posted August 2013.
 - * Removal of severity levels to become a short recommendation to reduce the risk of the discussed threat.
- o Version 04 Posted February 2013.
 - * Clear statement that the document compares threats of public LISP deployments with threats in the current Internet architecture.
 - * Addition of a severity level discussion at the end of each section.
 - * Addressed comments from V. Ermagan and D. Lewis' reviews.
 - * Updated References.
 - * Further editorial polishing.
- o Version 03 Posted October 2012.
 - * Dropped Reference to RFC 2119 notation because it is not actually used in the document.
 - * Deleted future plans section.
 - * Updated References
 - * Deleted/Modified sentences referring to the early status of the LISP WG and documents at the time of writing early versions of the document.

- * Further editorial polishing.
- * Fixed all ID nits.
- o Version 02 Posted September 2012.
 - * Added a new attack that combines over-claiming and de-aggregation (see Section 3.8).
 - * Editorial polishing.
- o Version 01 Posted February 2012.
 - * Added discussion on LISP-DDT.
- o Version 00 Posted July 2011.
 - * Added discussion on LISP-MS>.
 - * Added discussion on Instance ID.
 - * Editorial polishing of the whole document.
 - * Added "Change Log" appendix to keep track of main changes.
 - * Renamed "draft-saucez-lisp-security-03.txt".

Authors' Addresses

Damien Saucez
INRIA
2004 route des Lucioles BP 93
06902 Sophia Antipolis Cedex
France

Email: damien.saucez@inria.fr

Luigi Iannone
Telecom ParisTech
23, Avenue d'Italie, CS 51327
75214 PARIS Cedex 13
France

Email: ggx@gigix.net

Olivier Bonaventure
Universite catholique de Louvain
Place St. Barbe 2
Louvain la Neuve
Belgium

Email: olivier.bonaventure@uclouvain.be

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 4, 2014

D. Lewis
Cisco Systems, Inc.
P. Agarwal
Broadcom
L. Kreeger
F. Maino
P. Quinn
Cisco Systems, Inc.
M. Smith
N. Yadav
Insieme Networks
October 1, 2013

LISP Generic Protocol Extension
draft-lewis-lisp-gpe-01.txt

Abstract

This draft describes a mechanism for adding generalized multi-protocol support to the Locator/ID Separation Protocol (LISP) [RFC6830]. Protocol identification is carried in the LISP header and is used to describe the encapsulated payload.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 4, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. LISP Header	4
3. Generic Protocol Extension for LISP (LISP-gpe)	5
3.1. LISP-gpe Header	5
4. Backward Compatibility	6
4.1. LISP-gpe Routers to (legacy) LISP Routers	6
4.2. (legacy) LISP Routers to LISP-gpe Routers	6
4.3. Type of Service	6
4.4. VLAN Identifier (VID)	6
5. LISP-gpe Examples	7
6. Security Considerations	9
7. Acknowledgments	10
8. IANA Considerations	11
9. References	12
9.1. Normative References	12
9.2. Informative References	12
Authors' Addresses	13

1. Introduction

LISP [RFC6830] defines an encapsulation format that carries IPv4 or IPv6 (henceforth referred to as IP) packets in a LISP header and outer UDP/IP transport. The LISP header does not specify the protocol being encapsulated and therefore is currently limited to encapsulating only IP packet payloads. Other protocols, most notably VXLAN [VXLAN] (which defines a similar header format to LISP), are used to encapsulate L2 protocols such as Ethernet. LISP [RFC6830] can be extended to indicate the inner protocol, enabling the encapsulation of Ethernet, IP or any other desired protocol all the while ensuring compatibility with existing LISP [RFC6830] deployments.

This document describes extending LISP ([RFC6830]) to support additional payload types beyond IP packets. To support this capability, two elements of the existing LISP header are modified.

1. A flag bit is allocated, and set in the LISP header.
2. A 16 bit Protocol Type field is present in the LISP header.

These changes allow for the LISP header to support many different types of payloads. Backward compatibility with existing LISP tunnel routers is discussed in section 4.

2. LISP Header

As described in the introduction, the LISP header has no protocol identifier that indicates the type of payload being carried by LISP. Because of this, LISP is limited to an IP payload.

The LISP header contains flags (some defined, some reserved), a Nonce/Map-version field and an instance ID/Locator-status-bit field. The flags provide flexibility to define how the reserved bits can be used to change the definition of the LISP header.

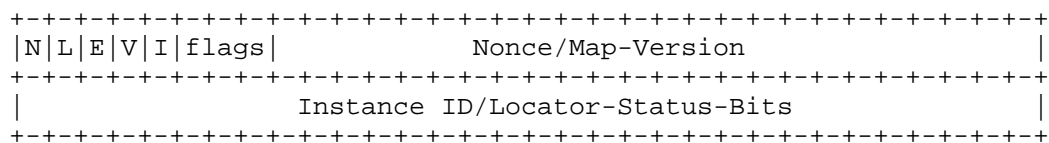


Figure 1: LISP Header

3. Generic Protocol Extension for LISP (LISP-gpe)

3.1. LISP-gpe Header

This draft defines two changes to the LISP header in order to support multi-protocol encapsulation.

P Bit: Flag bit 5 is defined as the P bit. The P bit **MUST** be set to 1 to indicate the presence of the 16 bit protocol type field in the lower 16 bits of the first word.

P = 0 indicates that the payload **MUST** conform to LISP as defined in [RFC6830].

Flag bit 5 was chosen as the P bit because this flag bit is currently unallocated in LISP [RFC6830].

Protocol Type Field: The lower 16 bits of the first word are used to carry a protocol type. This protocol type field contains the protocol, as defined in in [RFC1700] and in [ETYPES], of the encapsulated payload packet.

LISP [RFC6830] uses the lower 16 bits of the first word for either a nonce, an echo-nonce ([RFC6830]) or to support map-versioning ([RFC6834]). These are all optional capabilities that are indicated by setting the N, E, and the V bit respectively.

To maintain the desired data plane compatibility, when the P bit is set, the N, E, and V bits **MUST** be set to zero.

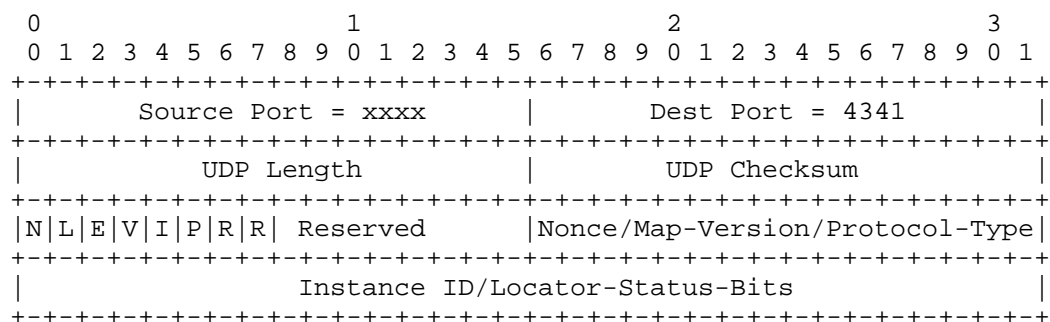


Figure 2: UDP + LISP-gpe

4. Backward Compatibility

An undefined (in RFC6830) flag bit, 5, was selected to ensure compatibility with existing LISP [RFC6830] deployments.

Similarly, using P = 0 to indicate that the format of the header and payload conforms to [RFC6830] ensures compatibility with existing LISP hardware forwarding platforms.

4.1. LISP-gpe Routers to (legacy) LISP Routers

A LISP-gpe router MUST not encapsulate non-IP packets to a LISP router. A method for determining the capabilities of a LISP router (gpe or "legacy") is out of the scope of this draft.

When encapsulating IP packets to a LISP router the P bit SHOULD be set to 1 and the UDP port MUST be set to 4341. The Protocol Type field SHOULD be 0x800 or 0x86DD. The (legacy) LISP router will ignore the P bit and the protocol type field. The (legacy) LISP router will treat the packet as a LISP packet and inspect the first nibble of the payload to determine the IP version.

When the P bit is set, the N, E, and V bits MUST be set to zero. The receiving (legacy) LISP router will ignore N, E and V bits, when the P bit is set.

4.2. (legacy) LISP Routers to LISP-gpe Routers

When a LISP-gpe router receives a packet from a (legacy) LISP router, the P bit MUST not be set and the UDP port MUST be 4341. The payload MUST be IP, and the LISP-gpe router will inspect the first nibble of the payload to determine IP version.

4.3. Type of Service

When a LISP-gpe router performs Ethernet encapsulation, the inner 802.1Q [IEEE8021Q] priority code point (PCP) field MAY be mapped from the encapsulated frame to the Type of Service field in the outer IPv4 header, or in the case of IPv6 the 'Traffic Class' field.

4.4. VLAN Identifier (VID)

When a LISP-gpe router performs Ethernet encapsulation, the inner header 802.1Q [IEEE8021Q] VLAN Identifier (VID) MAY be mapped to, or used to determine the LISP Instance ID field.

5. LISP-gpe Examples

This section provides two examples of IP protocols, and one example of Ethernet encapsulated LISP-gpe using the generic extension described in this document.

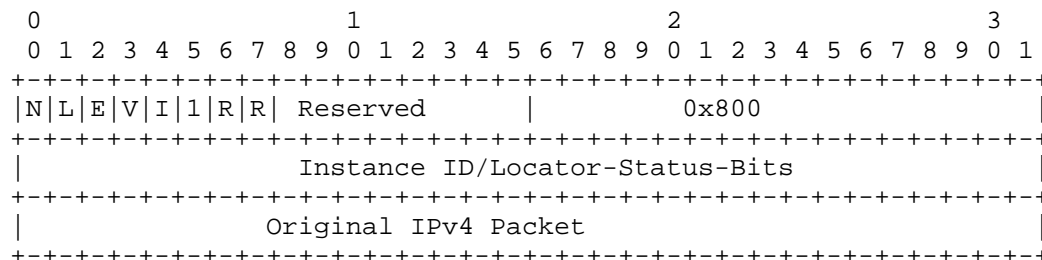


Figure 3: IPv4 and LISP-gpe

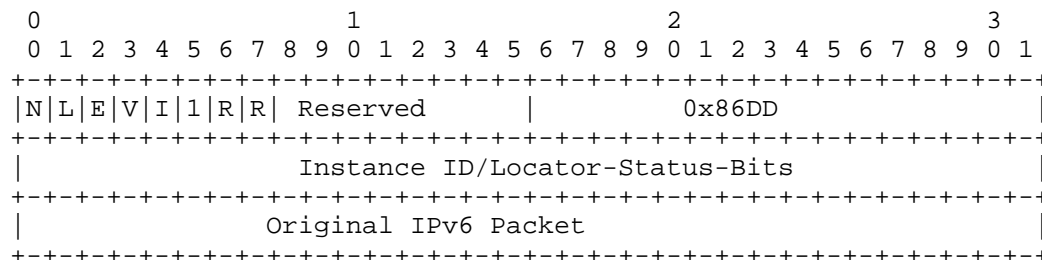


Figure 4: IPv6 and LISP-gpe

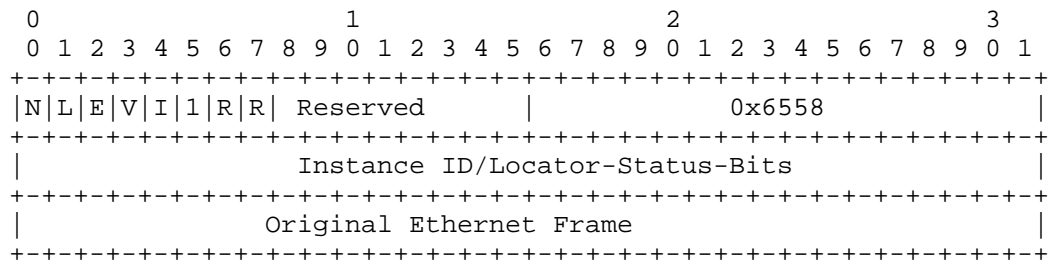


Figure 5: Ethernet and LISP-gpe

6. Security Considerations

LISP-gpe security considerations are similar to the LISP security considerations documented at length in LISP [RFC6830]. With LISP-gpe, issues such as dataplane spoofing, flooding, and traffic redirection are dependent on the particular protocol payload encapsulated.

7. Acknowledgments

A special thank you goes to Dino Farinacci for his guidance and detailed review.

8. IANA Considerations

This document creates no new requirements on IANA namespaces [RFC5226].

9. References

9.1. Normative References

- [RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.
- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

9.2. Informative References

- [ETYPES] The IEEE Registration Authority, "IEEE 802 Numbers", 2012, <<http://www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.xml>>.
- [IEEE8021Q] The IEEE Computer Society, "Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks", August 2012, <<http://standards.ieee.org/getieee802/download/802.1Q-2011.pdf>>.
- [RFC1700] Reynolds, J. and J. Postel, "Assigned Numbers", RFC 1700, October 1994.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [VXLAN] Dutt, D., Mahalingam, M., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", 2013.

Authors' Addresses

Darrel Lewis
Cisco Systems, Inc.

Email: darlewis@cisco.com

Puneet Agarwal
Broadcom

Email: pagarwal@broadcom.com

Larry Kreeger
Cisco Systems, Inc.

Email: kreeger@cisco.com

Fabio Maino
Cisco Systems, Inc.

Email: kreeger@cisco.com

Paul Quinn
Cisco Systems, Inc.

Email: paulq@cisco.com

Michael Smith
Insieme Networks

Email: michsmit@insiemenetworks.com

Navindra Yadav
Insieme Networks

Email: nyadav@insiemenetworks.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: June 18, 2018

D. Lewis
Cisco
J. Lemon
Broadcom
P. Agarwal
Innovium
L. Kreeger

P. Quinn
M. Smith
N. Yadav
F. Maino, Ed.
Cisco

December 15, 2017

LISP Generic Protocol Extension
draft-lewis-lisp-gpe-04

Abstract

This draft describes extending the Locator/ID Separation Protocol (LISP), via changes to the LISP header, to support multi-protocol encapsulation.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 18, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Conventions	3
1.2. Definition of Terms	3
2. LISP Header Without Protocol Extensions	3
3. Generic Protocol Extension for LISP (LISP-GPE)	3
4. Backward Compatibility	5
4.1. Type of Service	5
4.2. VLAN Identifier (VID)	5
5. IANA Considerations	5
6. Security Considerations	6
7. Acknowledgements	6
8. References	6
8.1. Normative References	6
8.2. Informative References	7
Authors' Addresses	7

1. Introduction

LISP, as defined in [RFC6830] and extended in [I-D.ietf-lisp-rfc6830bis], defines an encapsulation format that carries IPv4 or IPv6 (henceforth referred to as IP) packets in a LISP header and outer UDP/IP transport.

The LISP header does not specify the protocol being encapsulated and therefore is currently limited to encapsulating only IP packet payloads. Other protocols, most notably VXLAN [RFC7348] (which defines a similar header format to LISP), are used to encapsulate L2 protocols such as Ethernet.

This document defines an extension for the LISP header, as defined in [I-D.ietf-lisp-rfc6830bis], to indicate the inner protocol, enabling the encapsulation of Ethernet, IP or any other desired protocol all the while ensuring compatibility with existing LISP deployments.

A flag in the LISP header, called the P-bit, is used to signal the presence of the 8-bit Next Protocol field. The Next Protocol field,

when present, uses 8 bits of the field allocated to the echo-noncing and map-versioning features. The two features are still available, albeit with a reduced length of Nonce and Map-Version.

1.1. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

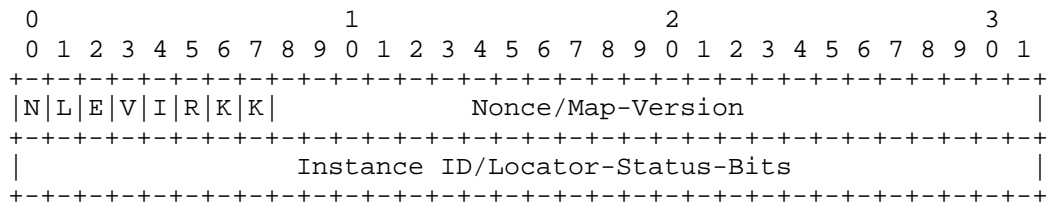
1.2. Definition of Terms

This document uses terms already defined in [I-D.ietf-lisp-rfc6830bis].

2. LISP Header Without Protocol Extensions

As described in the introduction, the LISP header has no protocol identifier that indicates the type of payload being carried. Because of this, LISP is limited to carry IP payloads.

The LISP header [I-D.ietf-lisp-rfc6830bis] contains a series of flags (some defined, some reserved), a Nonce/Map-version field and an instance ID/Locator-status-bit field. The flags provide flexibility to define how the various fields are encoded. Notably, Flag bit 5 is the last reserved bit in the LISP header.



LISP Header

3. Generic Protocol Extension for LISP (LISP-GPE)

This document defines the following changes to the LISP header in order to support multi-protocol encapsulation:

P Bit: Flag bit 5 is defined as the Next Protocol bit. The P bit MUST be set to 1 to indicate the presence of the 8 bit next protocol field.

P = 0 indicates that the payload MUST conform to LISP as defined in [I-D.ietf-lisp-rfc6830bis]. Flag bit 5 was chosen as the P bit because this flag bit is currently unallocated.

Next Protocol: The lower 8 bits of the first 32-bit word are used to carry a Next Protocol. This Next Protocol field contains the protocol of the encapsulated payload packet.

LISP uses the lower 24 bits of the first word for either a nonce, an echo-nonce, or to support map-versioning [RFC6834]. These are all optional capabilities that are indicated in the LISP header by setting the N, E, and the V bit respectively.

When the P-bit and the N-bit are set to 1, the Nonce field is the middle 16 bits.

When the P-bit and the V-bit are set to 1, the Version field is the middle 16 bits.

When the P-bit is set to 1 and the N-bit and the V-bit are both 0, the middle 16-bits are set to 0.

This draft defines the following Next Protocol values:

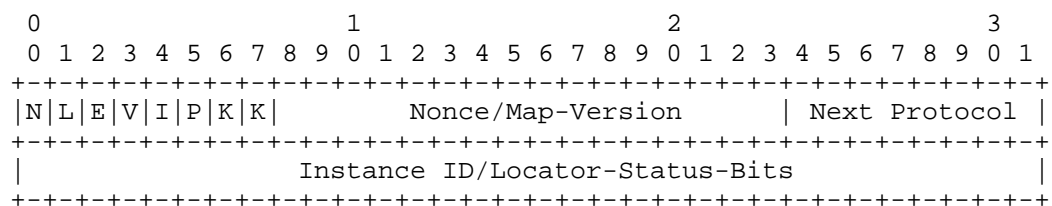
0x1 : IPv4

0x2 : IPv6

0x3 : Ethernet

0x4 : Network Service Header [I-D.ietf-sfc-nsh]

0x6: Group-Based Policy (GBP) [I-D.lemon-vxlan-gpe-gbp].



LISP-GPE Header

4. Backward Compatibility

LISP-GPE uses the same UDP destination port (4341) allocated to LISP.

A LISP-GPE router **MUST** not encapsulate non-IP packets to a LISP router. A method for determining the capabilities of a LISP router (GPE or "legacy") is out of the scope of this draft.

When encapsulating IP packets to a LISP "legacy" router the P bit **MUST** be set to 0.

4.1. Type of Service

When a LISP-GPE router performs Ethernet encapsulation, the inner 802.1Q [IEEE8021Q] priority code point (PCP) field **MAY** be mapped from the encapsulated frame to the Type of Service field in the outer IPv4 header, or in the case of IPv6 the 'Traffic Class' field.

4.2. VLAN Identifier (VID)

When a LISP-GPE router performs Ethernet encapsulation, the inner header 802.1Q [IEEE8021Q] VLAN Identifier (VID) **MAY** be mapped to, or used to determine the LISP Instance ID field.

5. IANA Considerations

IANA is requested to set up a registry of LISP-GPE "Next Protocol". These are 8-bit values. Next Protocol values in the table below are defined in this draft. New values are assigned via Standards Action [RFC5226].

Next Protocol	Description	Reference
0	Reserved	This Document
1	IPv4	This Document
2	IPv6	This Document
3	Ethernet	This Document
4	NSH	This Document
5	Reserved	
6	GBP	This Document
7	Reserved	
8..255	Unassigned	

6. Security Considerations

LISP-GPE security considerations are similar to the LISP security considerations documented at length in [I-D.ietf-lisp-rfc6830bis]. With LISP-GPE, issues such as dataplane spoofing, flooding, and traffic redirection may depend on the particular protocol payload encapsulated.

7. Acknowledgements

A special thank you goes to Dino Farinacci for his guidance and detailed review.

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <<https://www.rfc-editor.org/info/rfc5226>>.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <<https://www.rfc-editor.org/info/rfc6830>>.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <<https://www.rfc-editor.org/info/rfc6834>>.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <<https://www.rfc-editor.org/info/rfc7348>>.

8.2. Informative References

[I-D.ietf-lisp-rfc6830bis]

Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos-Aparicio, "The Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-rfc6830bis-07 (work in progress), November 2017.

[I-D.ietf-sfc-nsh]

Quinn, P., Elzur, U., and C. Pignataro, "Network Service Header (NSH)", draft-ietf-sfc-nsh-28 (work in progress), November 2017.

[I-D.lemon-vxlan-gpe-gbp]

Lemon, J., Maino, F., and M. Smith, "Group Policy Encoding with VXLAN-GPE", draft-lemon-vxlan-gpe-gbp-00 (work in progress), October 2017.

Authors' Addresses

Darrel Lewis
Cisco Systems

Email: darlewis@cisco.com

John Lemon
Broadcom
3151 Zanker Road
San Jose, CA 95134
USA

Email: john.lemon@broadcom.com

Puneet Agarwal
Innovium
USA

Email: puneet@acm.org

Larry Kreeger
USA

Email: lkreeger@gmail.com

Paul Quinn
Cisco Systems

Email: pquinn@cisco.com

Michael Smith
Cisco Systems

Email: michsmit@cisco.com

Navindra Yadav
Cisco Systems

Email: nyadav@cisco.com

Fabio Maino (editor)
Cisco Systems
San Jose, CA 95134
USA

Email: fmaino@cisco.com

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: April 21, 2014

F. Maino
V. Ermagan
Y. Hertoghs
Cisco Systems
D. Farinacci
lispers.net
M. Smith
Insieme Networks
October 18, 2013

LISP Control Plane for Network Virtualization Overlays
draft-maino-nvo3-lisp-cp-03

Abstract

The purpose of this draft is to analyze the mapping between the Network Virtualization over L3 (NVO3) requirements and the capabilities of the Locator/ID Separation Protocol (LISP) control plane. This information is provided as input to the NVO3 analysis of the suitability of existing IETF protocols to the NVO3 requirements.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 21, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definition of Terms	4
3. LISP Overview	4
3.1. LISP Site Configuration	6
3.2. End System Provisioning	6
3.3. End System Registration	7
3.4. Packet Flow and Control Plane Operations	7
3.4.1. Supporting ARP Resolution with LISP Mapping System	8
3.5. End System Mobility	10
3.6. L3 LISP	12
4. Reference Model	12
4.1. LISP NVE Service Types	14
4.1.1. LISP L2 NVE Services	14
4.1.2. LISP L3 NVE Services	14
5. Functional Components	14
5.1. Generic Service Virtualization Components	14
5.1.1. Virtual Attachment Points (VAPs)	15
5.1.2. Overlay Modules and Tenant ID	15
5.1.3. Tenant Instance	15
5.1.4. Tunnel Overlays and Encapsulation Options	16
5.1.5. Control Plane Components	16
6. Key Aspects of Overlay	17
6.1. Overlay Issues to Consider	17
6.1.1. Data Plane vs. Control Plane Driven	17
6.1.2. Data Plane and Control Plane Separation	17
6.1.3. Handling Broadcast, Unknown Unicast and Multicast (BUM) Traffic	17
7. Security Considerations	18
8. IANA Considerations	18
9. Acknowledgements	18
10. References	18
10.1. Normative References	18
10.2. Informative References	18

Authors' Addresses	21
--------------------	----

1. Introduction

The purpose of this draft is to analyze the mapping between the Network Virtualization over L3 (NVO3) [I-D.ietf-nvo3-overlay-problem-statement] requirements and the capabilities of the Locator/ID Separation Protocol (LISP) [RFC6830] control plane. This information is provided as input to the NVO3 analysis of the suitability of existing IETF protocols to the NVO3 requirements.

LISP is a flexible map and encap framework that can be used for overlay network applications, including Data Center Network Virtualization.

The LISP framework provides two main tools for NVO3: (1) a Data Plane that specifies how Endpoint Identifiers (EIDs) are encapsulated in Routing Locators (RLOCs), and (2) a Control Plane that specifies the interfaces to the LISP Mapping System that provides the mapping between EIDs and RLOCs.

This document focuses on the control plane for L2 over L3 LISP encapsulation, where EIDs are associated with MAC addresses. As such the LISP control plane can be used with the data path encapsulations defined in VXLAN [I-D.mahalingam-dutt-dcops-vxlan] and in NVGRE [I-D.sridharan-virtualization-nvgre]. The LISP control plane can, of course, be used with the L2 LISP data path encapsulation defined in [I-D.smith-lisp-layer2].

The LISP control plane provides the Mapping Service for the Network Virtualization Edge (NVE), mapping per-tenant end system identity information on the corresponding location at the NVE. As required by NVO3, LISP supports network virtualization and tenant separation to hide tenant addressing information, tenant-related control plane activity and service contexts from the underlay network.

The LISP control plane is extensible, and can support non-LISP data path encapsulations such as [I-D.sridharan-virtualization-nvgre], or other encapsulations that provide support for network virtualization. [RFC6832] specifies an open interworking framework to allow LISP to non-LISP sites communication.

Broadcast, unknown unicast, and multicast in the overlay network are supported by either replicated unicast, or core-based multicast as specified in [RFC6831], [I-D.farinacci-lisp-mr-signaling], and [I-D.farinacci-lisp-te].

Finally, the LISP architecture has a modular design that allows the use of different Mapping Databases, provided that the interface to the Mapping System remains the same [RFC6833]. This allows for different Mapping Databases that may fit different NVO3 deployments. As an example of the modularity of the LISP Mapping System, a worldwide LISP pilot network is currently using an hierarchical Delegated Database Tree [I-D.ietf-lisp-ddt], after having been operated for years with an overlay BGP mapping infrastructure [RFC6836].

The LISP mapping system supports network virtualization, and a single mapping infrastructure can run multiple instances, either public or private, of the mapping database.

The rest of this document, after giving a quick a LISP overview in Section 3, follows the functional model defined in [I-D.ietf-nvo3-framework] that provides in Section 4 an overview of the LISP NVO3 reference model, and in Section 5 a description of its functional components. Section 6 contains various considerations on key aspects of LISP NVO3, followed by security considerations in Section 7.

2. Definition of Terms

Flood-and-Learn: the use of dynamic (data plane) learning in VXLAN to discover the location of a given Ethernet/IEEE 802 MAC address in the underlay network.

ARP-Agent Reply: the ARP proxy-reply of an agent (e.g. an ITR) with a MAC address of some other system in response to an ARP request to a target which is not the agent's IP address

For definition of NVO3 related terms, notably Virtual Network (VN), Virtual Network Identifier (VNI), Network Virtualization Edge (NVE), Data Center (DC), please consult [I-D.ietf-nvo3-framework].

For definitions of LISP related terms, notably Map-Request, Map-Reply, Ingress Tunnel Router (ITR), Egress Tunnel Router (ETR), Map-Server (MS) and Map-Resolver (MR) please consult the LISP specification [RFC6830].

3. LISP Overview

This section provides a quick overview of L2 LISP, with focus on control plane operations.

The modular and extensible architecture of the LISP control plane allows its use with both L2 or L3 LISP data path encapsulation. In

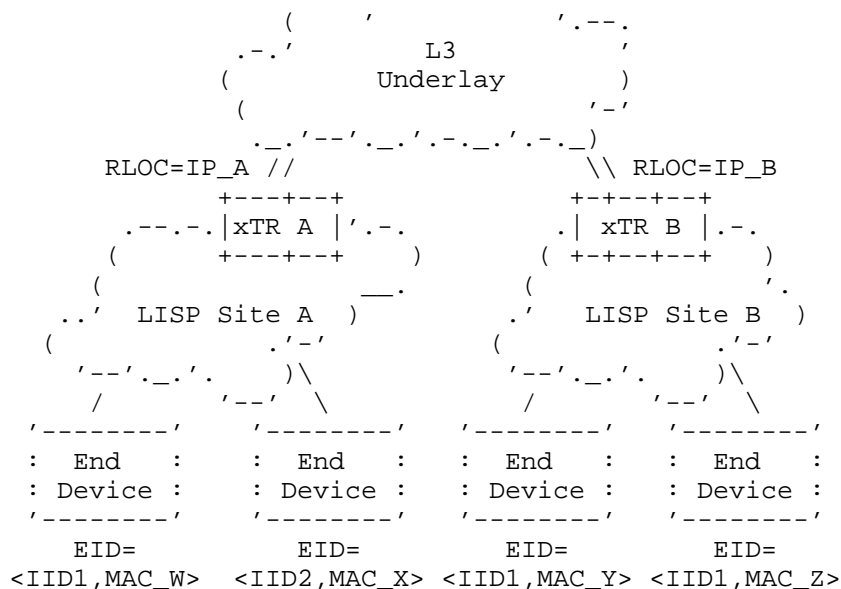


Figure 1: Example of L2 NV03 Services

3.1. LISP Site Configuration

In each LISP site the xTRs are configured with an IP address (the site RLOCs) per each interface facing the underlay network.

Similarly the MS/MR are assigned an IP address in the RLOC space.

The configuration of the xTRs includes the RLOCs of the MS/MR and a shared secret that is optionally used to secure the communication between xTRs and MS/MR.

To provide support for multi-tenancy multiple instances of the mapping database are identified by a LISP Instance ID (IID), that is equivalent to the 24-bit VXLAN Network Identifier (VNI) or Tenant Network Identifier (TNI) that identifies tenants in [I-D.mahalingam-dutt-dcops-vxlan].

3.2. End System Provisioning

We assume that a provisioning framework will be responsible for provisioning end systems (e.g. VMs) in each data center. The provisioning system configures each end system with an Ethernet/IEEE 802 MAC address and/or IP address and provisions the NVE with other end system specific attributes such as VLAN information, and TS/VLAN to VNI mapping information. LISP does not introduce new addressing requirements for end systems.

The provisioning infrastructure is also responsible to provide a network attach function, that notifies the network virtualization edge (the LISP site ETR) that the end system is attached to a given virtual network (identified by its VNI/IID) and that the end system is identified, within that virtual network, by a given Ethernet/IEEE 802 MAC address.

3.3. End System Registration

Upon notification of end system network attach, that includes the EID=<IID,MAC> tuple that identifies that end system, the ETR sends a LISP Map-Register to the Mapping System. The Map-Register includes the EID and RLOCs of the LISP site. The EID-to-RLOC mapping is now available, via the Mapping System Infrastructure, to other LISP sites that are hosting end systems that belong to the same tenant.

For more details on end system registration see [RFC6833].

3.4. Packet Flow and Control Plane Operations

This section provides an example of the unicast packet flow and the control plane operations when in the topology shown in Figure 1 end system W, in LISP site A, wants to communicate to end system Y in LISP site B. We'll assume that W knows Y's EID MAC address (e.g. learned via ARP).

- o W sends an Ethernet/IEEE 802 MAC frame with destination EID=<IID1,MAC_Y> and source EID=<IID1,MAC_W>.
- o ITR A does a lookup in its local map-cache for the destination EID=<IID1, MAC_Y>. Since this is the first packet sent to MAC_Y, the map-cache is a miss, and the ITR sends a Map-request to the mapping database system looking up the EID=<IID1,MAC_Y>.
- o The mapping systems forwards the Map-Request to ETR B, that is aware of the EID-to-RLOC mapping for <IID1,MAC_Y>. Alternatively, depending on the mapping system configuration, a Map-Server which is part of the mapping database system may send a Map-Reply directly to ITR A.

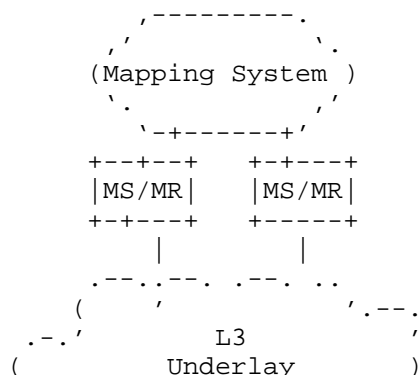
- o ETR B sends a Map-Reply to ITR A that includes the EID-to-RLOC mapping: EID=<IID1,MAC_Y> -> RLOC=IP_B, where IP_B is the locator of ETR B, hence the locator of LISP site B. In order to facilitate interoperability, the Map-Reply may also include attributes such as the data plane encapsulations supported by the ETR.
- o ITR A populates the local map-cache with the EID to RLOC mapping, and either L2 LISP, VXLAN, or NVGRE encapsulates all subsequent packets with a destination EID=<IID1,MAC_Y> with a destination RLOC=IP_B.

It should be noted how the LISP mapping system replaces the use of Flood-and-Learn based on multicast distribution trees instantiated in the underlay network (required by VXLAN's dynamic data plane learning), with a unicast control plane and a cache mechanism that "pulls" on-demand the EID-to-RLOC mapping from the LISP mapping database. This improves scalability, and simplifies the configuration of the underlay network.

3.4.1. Supporting ARP Resolution with LISP Mapping System

A large majority of data center applications are IP based, and in those use cases end systems are provisioned with IP addresses as well as MAC addresses.

In this case, to eliminate the flooding of ARP traffic and further reduce the need for multicast in the underlay network, the LISP mapping system is used to support ARP resolution at the ITR. We assume that as shown in Figure 2: (1) end system W has an IP address IP_W, and end system Y has an IP address IP_Y, (2) end system W knows Y's IP address (e.g. via DNS lookup). We also assume that during registration Y has registered both its MAC address and its IP address as EID. End system Y is then identified by the EID = <IID1,IP_Y,MAC_Y>.



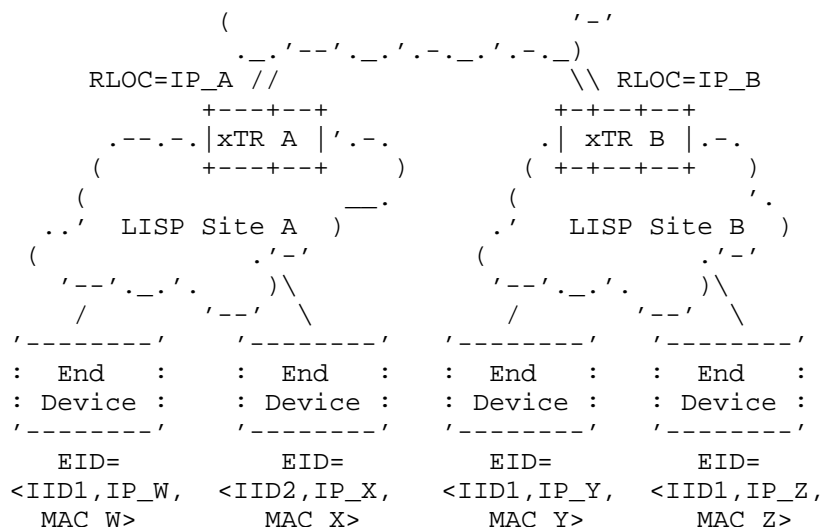


Figure 2: Example of L3 NVO3 Services

The packet flow and control plane operation are as follows:

- o End system W sends a broadcast ARP message to discover the MAC address of end system Y. The message contains IP_Y in the ARP message payload.
- o ITR A, acting as a L2 switch, will receive the ARP message, but rather than flooding it on the overlay network sends a Map-Request to the mapping database system for EID = <IID1,IP_Y,*>.
- o The Map-Request is routed by the mapping system infrastructure to ETR B, that will send a Map-Reply back to ITR A containing the mapping EID=<IID1,IP_Y,MAC_Y> -> RLOC=IP_B, (the locator of ETR B). Alternatively, depending on the mapping system configuration, a Map-Server in the mapping system may send directly a Map-Reply to ITR A.
- o ITR A populates the map-cache with the received entry, and sends an ARP-Agent Reply to W that includes MAC_Y and IP_Y.
- o End system W learns MAC_Y from the ARP message and can now send a packet to end system Y by including MAC_Y, and IP_Y, as destination addresses.
- o ITR A will then process the packet as specified in Section 3.4.

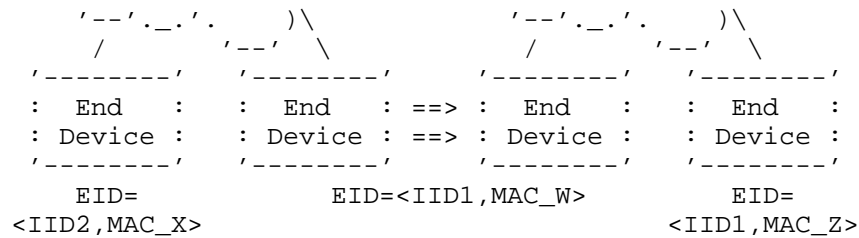


Figure 3: End System Mobility

As a result of the end system registration, described in Section 3.3, the Mapping System contains the EID-to-RLOC mapping for end system W that associates EID=<IID1, MAC_W> with the RLOC(s) associated with LISP site A (IP_A).

The process of migrating end system W from data center A to data center B is initiated.

ETR B receives a pre-associate message that includes EID=<IID1, MAC_W>. ETR B sends a Map-Register to the mapping system registering RLOC=IP_B as an additional locator for end system W with priority set to 255. This means that the RLOC MUST NOT be used for unicast forwarding, but the mapping system is now aware of the new location.

During the migration process of end system W, ETR A receives a dissociate message, and sends a Map-Register with Record TTL=0 to signal the mapping system that end system W is no longer reachable at RLOC=IP_A. ETR A will also add an entry in its forwarding table that marks EID=<IID1, MAC_W> as non-local. When end system W has completed its migration, ETR B receives an associate message for end system W, and sends a Map-Register to the mapping system setting a non-255 priority for RLOC=IP_B. Now the mapping system is updated with the new EID-to-RLOC mapping for end system W with the desired priority.

The remote ITRs that were corresponding with end system W during the migration will keep sending packets to ETR A. ETR A will keep forwarding locally those packets until it receives a dissociate message, and the entry in the forwarding table associated with EID=<IID1, MAC_W> is marked as non-local. Subsequent packets arriving at ETR A from a remote ITR, and destined to end system W will hit the entry in the forwarding table that will generate an exception, and will generate a Solicit-Map-Request (SMR) message that is returned to the remote ITR. Upon receiving the SMR the remote ITR will invalidate its local map-cache entry for EID=<IID1, MAC_W> and send a new Map-Request for that EID. The Map-Request will generate a Map-Reply that includes the new EID-to-RLOC mapping for end system W

with RLOC=IP_B. Similarly, unencapsulated packets arriving at ITR A from local end systems and destined to end system W will hit the entry in the forwarding table marked as non-local, and will generate an exception that by sending a Map-Request for EID=<IID1, MAC_W> will populate the map-cache of ITR A with an EID-to-RLOC mapping for end system W with RLOC=IP_B.

3.6. L3 LISP

The two examples above shows how the LISP control plane can be used in combination with either L2 LISP, VXLAN, or NVGRE encapsulation to provide L2 network virtualization services across data centers.

There is a trend, led by Massive Scalable Data Centers, that is accelerating the adoption of L3 network services in the data center, to preserve the many benefits introduced by L3 (scalability, multi-homing, ...).

LISP, as defined in [RFC6830], provides L3 network virtualization services over an L3 underlay network that, as an alternative to L2 overlay solutions, matches the requirements for DC Network Virtualization. L2 overlay solutions are necessary for Data Centers that rely on non IPv4/IPv6 protocols, but when IP is pervasive L3 LISP provides a better and more scalable overlay.

4. Reference Model

Figure 4, taken from [I-D.ietf-nvo3-framework], introduces the NVO3 reference model.

In a LISP NVO3 network the Tenant Systems (TS) that are homed to a common NVE, having specific Endpoint Identifiers (EIDs), are part of a 'LISP site'.

The network virtualization edge (NVE) function is performed by Ingress Tunnel Routers (ITRs) that are responsible for encapsulating the LISP ingress traffic, and Egress Tunnel Routers (ETRs) that are responsible for de-encapsulating the LISP egress traffic.

The outer tunnel IP addresses (either IPv4 or IPv6) on the ITR and ETR NVE function are known as Routing Locators (RLOCs).

ETRs are also responsible to register the EID-to-RLOC mapping for a given LISP site in the LISP mapping database system [RFC6833] .

ITRs and ETRs, collectively referred as xTRs, provide for tenant separation, perform the encap/decap function, and interface with the LISP Mapping System that maps tenant addressing information (in the

EID name space) on the underlay L3 infrastructure (in the RLOC name space), with the encoding defined in [I-D.ietf-lisp-lcaf].

The LISP Mapping system is a distributed mapping infrastructure, accessible via Map Servers (MS) and Map Resolvers (MR), that performs the NVA function.

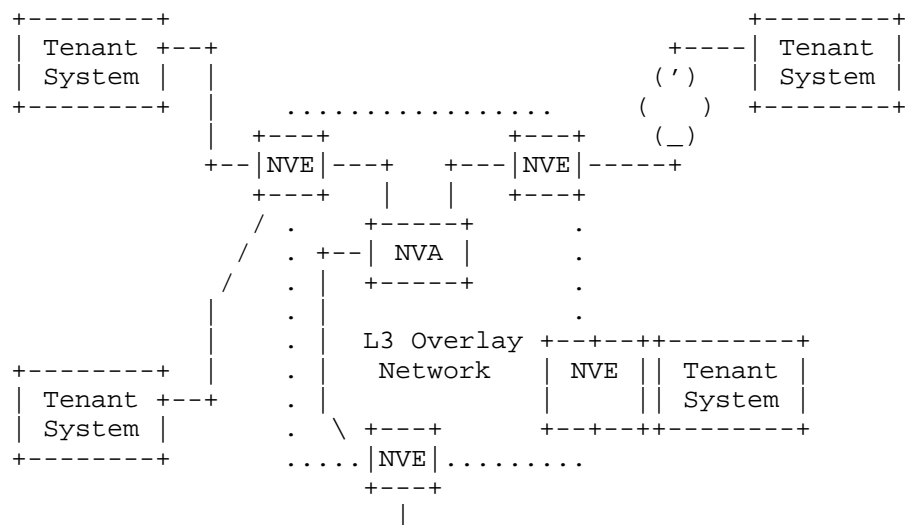
The LISP Mapping system can be scaled across various physical components e.g. across an EID based hierarchy as described in [I-D.ietf-lisp-ddt]. EID prefixes and/or address families can thus be scaled across the mapping infrastructure if needed.

The NVA function can offer a northbound SDN interface in order to program the EID to RLOC mapping from e.g. an orchestration system. An example is given in [I-D.barkai-lisp-nfv] .

As traffic reaches An ingress NVE, the corresponding ITR uses the LISP Map-Request/Reply service to determine the location of the destination End System.

The LISP mapping system combines the distribution of address advertisement and (stateless) tunneling provisioning.

LISP defines several mechanisms for determining RLOC reachability, including Locator Status Bits, "nonce echoing", and RLOC probing. Please see Sections 5.3 and 6.3 of [RFC6830]. However, given the fact that DC's are typically deployed with a single stage IGP hierarchy, the IGP responsible for the RLOC space offers enough reachability information.



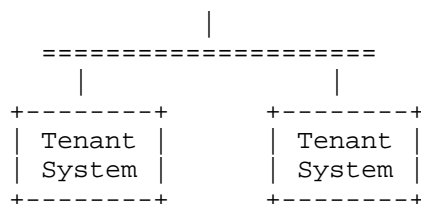


Figure 4: NV03 Generic Reference Model

4.1. LISP NVE Service Types

L2 NVE and L3 NVE service types thanks to the flexibility provided by the LISP Canonical Address Format [I-D.ietf-lisp-lcaf], that allows for EIDs to be encoded either as MAC addresses or IP addresses.

4.1.1. LISP L2 NVE Services

The frame format defined in [I-D.mahalingam-dutt-dcops-vxlan], has a header compatible with the LISP data path encapsulation header, when MAC addresses are used as EIDs, as described in section 4.12.2 of [I-D.ietf-lisp-lcaf].

The LISP control plane is extensible, and can support non-LISP data path encapsulations such as NVGRE [I-D.sridharan-virtualization-nvgre], or other encapsulations that provide support for network virtualization.

4.1.2. LISP L3 NVE Services

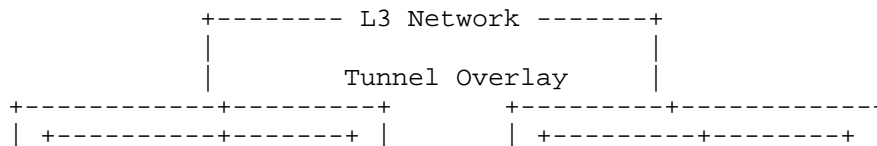
LISP is defined as a virtualized IP routing and forwarding service in [RFC6830], and as such can be used to provide L3 NVE services.

5. Functional Components

This section describes the functional components of a LISP NVE as defined in Section 3 of [I-D.ietf-nvo3-framework].

5.1. Generic Service Virtualization Components

The generic reference model for NVE is depicted in [I-D.ietf-nvo3-framework].



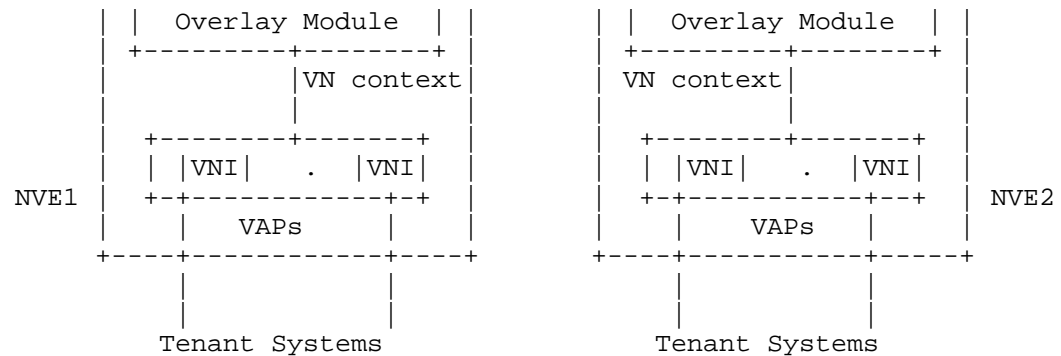


Figure 5: Generic reference model for NV Edge

5.1.1. Virtual Attachment Points (VAPs)

In a LISP NVE, Tunnel Routers (xTRs) implement the NVE functionality on ToRs or Virtual Switches. Tenant Systems attach to the Virtual Access Points (VAPs) provided by the xTRs (either a physical port or a virtual interface).

The VAPs are identified by either a physical port or a virtual interface, e.g. Indexed by VLAN tags, a set, range, or set of ranges of VLAN tags in the case of a L2 service, or a virtual routed interface Indexed by a VLAN in case of a L3 service, or a combination of them in case of An L2/L3 service.

5.1.2. Overlay Modules and Tenant ID

The xTR also implements the function of NVE Overlay Module, by mapping the addressing information (EIDs) of the tenant packet on the appropriate locations (RLOCs) in the underlay network. The Virtual Network Identifier (VNI) is encoded in the encapsulated packet (either in the 24-bit IID field of the LISP header for L2/L3 LISP encapsulation, or in the 24-bit VXLAN Network Identifier field for VXLAN encapsulation, or in the 24-bit NVGRE Tenant Network Identifier field of NVGRE). In a LISP NVE globally unique (per administrative domain) VNIs are used to identify the Tenant instances.

The mapping of the tenant packet address onto the underlay network location is "pulled" on-demand from the mapping system, and cached at the NVE in a per-VNI map-cache.

5.1.3. Tenant Instance

Tenants are mapped on LISP Instance IDs (IIDs), and the LISP Control Plane uses the IID to provide segmentation and virtualization. The

ETR is responsible to register the Tenant System to the LISP mapping system, via the Map-Register service provided by LISP Map-Servers (MS). The Map-Register includes the IID that is used to identify the tenant.

5.1.4. Tunnel Overlays and Encapsulation Options

The LISP control protocol, as defined today, provides support for L2 LISP and VXLAN L2 over L3 encapsulation, and LISP L3 over L3 encapsulation, as well as support for the Generic Protocol Extensions for LISP and VXLAN defined in [I-D.lewis-lisp-gpe] and [I-D.quinn-vxlan-gpe] respectively. The Generic Protocol Extensions can be used to offer a concurrent L2 and L3 overlay across the same dataplane.

We believe that the LISP control Protocol can be easily extended to support different IP tunneling options (such as NVGRE).

5.1.5. Control Plane Components

5.1.5.1. Auto-provisioning/Service Discovery

The LISP framework does not include mechanisms to provision the local NVE with the appropriate Tenant Instance for each Tenant System. Other protocols, such as VDP (in IEEE P802.1Qbg), should be used to implement a network attach/detach function.

The LISP control plane can take advantage of such a network attach/detach function to trigger the registration of a Tenant End System to the Mapping System. This is particularly helpful to handle mobility across DC of the Tenant End System.

It is possible to extend the LISP control protocol to advertise the tenant service instance (tenant and service type provided) to other NVEs, and facilitate interoperability between NVEs that are using different service types.

5.1.5.2. Address Advertisement and Tunnel mapping

As traffic reaches an ingress NVE, the corresponding ITR uses the LISP Map-Request/Reply service to determine the location of the destination End System.

The LISP mapping system combines the distribution of address advertisement and (stateless) tunneling provisioning.

When EIDs are mapped on both IP addresses and MACs, the need to flood ARP messages at the NVE is eliminated resolving the issues with explosive ARP handling.

5.1.5.3. Tunnel Management

LISP defines several mechanisms for determining RLOC reachability, including Locator Status Bits, "nonce echoing", and RLOC probing. Please see Sections 5.3 and 6.3 of [RFC6830].

6. Key Aspects of Overlay

6.1. Overlay Issues to Consider

6.1.1. Data Plane vs. Control Plane Driven

The use of LISP control plane minimizes the need for multicast in the underlay network overcoming the scalability limitations of VXLAN dynamic data plane learning (Flood-and-Learn).

Multicast or ingress replication in the underlay network are still required, as specified in [RFC6831], [I-D.farinacci-lisp-mr-signaling], and [I-D.farinacci-lisp-te], to support broadcast, unknown, and multicast traffic in the overlay, but multicast in the underlay is no longer required (at least for IP traffic) for unicast overlay services.

6.1.2. Data Plane and Control Plane Separation

LISP introduces a clear separation between data plane and control plane functions. LISP modular design allows for different mapping databases, to achieve different scalability goals and to meet requirements of different deployments.

6.1.3. Handling Broadcast, Unknown Unicast and Multicast (BUM) Traffic

Packet replication in the underlay network to support broadcast, unknown unicast and multicast overlay services can be done by:

- o Ingress replication
- o Use of underlay multicast trees

[RFC6831] specifies how to map a multicast flow in the EID space during distribution tree setup and packet delivery in the underlay network. LISP-multicast doesn't require packet format changes in multicast routing protocols, and doesn't impose changes in the internal operation of multicast in a LISP site. The only operational

changes are required in PIM-ASM [RFC4601], MSDP [RFC3618], and PIM-SSM [RFC4607].

7. Security Considerations

[I-D.ietf-lisp-sec] defines a set of security mechanisms that provide origin authentication, integrity and anti-replay protection to LISP's EID-to-RLLOC mapping data conveyed via mapping lookup process. LISP-SEC also enables verification of authorization on EID-prefix claims in Map-Reply messages.

Additional security mechanisms to protect the LISP Map-Register messages are defined in [RFC6833].

The security of the Mapping System Infrastructure depends on the particular mapping database used. The [I-D.ietf-lisp-ddt] specification, as an example, defines a public-key based mechanism that provides origin authentication and integrity protection to the LISP DDT protocol.

8. IANA Considerations

This document has no IANA implications

9. Acknowledgements

The authors want to thank Victor Moreno and Paul Quinn for the early review, insightful comments and suggestions.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3618] Fenner, B. and D. Meyer, "Multicast Source Discovery Protocol (MSDP)", RFC 3618, October 2003.
- [RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.
- [RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, August 2006.

10.2. Informative References

- [I-D.barkai-lisp-nfv]
sbarkai@gmail.com, s., Farinacci, D., Meyer, D., Maino, F., and V. Ermagan, "LISP Based FlowMapping for Scaling NFV", draft-barkai-lisp-nfv-02 (work in progress), July 2013.
- [I-D.farinacci-lisp-mr-signaling]
Farinacci, D. and M. Napierala, "LISP Control-Plane Multicast Signaling", draft-farinacci-lisp-mr-signaling-03 (work in progress), September 2013.
- [I-D.farinacci-lisp-te]
Farinacci, D., Lahiri, P., and M. Kowal, "LISP Traffic Engineering Use-Cases", draft-farinacci-lisp-te-03 (work in progress), July 2013.
- [I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-01 (work in progress), March 2013.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-03 (work in progress), September 2013.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., Saucez, D., and O. Bonaventure, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-04 (work in progress), October 2012.
- [I-D.ietf-nvo3-dataplane-requirements]
Bitar, N., Lasserre, M., Balus, F., Morin, T., Jin, L., and B. Khasnabish, "NVO3 Data Plane Requirements", draft-ietf-nvo3-dataplane-requirements-01 (work in progress), July 2013.
- [I-D.ietf-nvo3-framework]
Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for DC Network Virtualization", draft-ietf-nvo3-framework-03 (work in progress), July 2013.
- [I-D.ietf-nvo3-nve-nva-cp-req]
Kreeger, L., Dutt, D., Narten, T., and D. Black, "Network Virtualization NVE to NVA Control Protocol Requirements", draft-ietf-nvo3-nve-nva-cp-req-00 (work in progress), July 2013.

- [I-D.ietf-nvo3-overlay-problem-statement]
Narten, T., Gray, E., Black, D., Fang, L., Kreeger, L.,
and M. Napierala, "Problem Statement: Overlays for Network
Virtualization", draft-ietf-nvo3-overlay-problem-
statement-04 (work in progress), July 2013.
- [I-D.kompella-nvo3-server2nve]
Kompella, K., Rekhter, Y., Morin, T., and D. Black,
"Signaling Virtual Machine Activity to the Network
Virtualization Edge", draft-kompella-nvo3-server2nve-02
(work in progress), April 2013.
- [I-D.lewis-lisp-gpe]
Lewis, D., Agarwal, P., Kreeger, L., Quinn, P., Smith, M.,
and N. Yadav, "LISP Generic Protocol Extension", draft-
lewis-lisp-gpe-01 (work in progress), October 2013.
- [I-D.mahalingam-dutt-dcops-vxlan]
Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger,
L., Sridhar, T., Bursell, M., and C. Wright, "VXLAN: A
Framework for Overlaying Virtualized Layer 2 Networks over
Layer 3 Networks", draft-mahalingam-dutt-dcops-vxlan-04
(work in progress), May 2013.
- [I-D.quinn-vxlan-gpe]
Quinn, P., Agarwal, P., Fernando, R., Lewis, D., Kreeger,
L., Smith, M., and N. Yadav, "Generic Protocol Extension
for VXLAN", draft-quinn-vxlan-gpe-01 (work in progress),
October 2013.
- [I-D.smith-lisp-layer2]
Smith, M., Dutt, D., Farinacci, D., and F. Maino, "Layer 2
(L2) LISP Encapsulation Format", draft-smith-lisp-
layer2-03 (work in progress), September 2013.
- [I-D.sridharan-virtualization-nvgre]
Sridharan, M., Greenberg, A., Wang, Y., Garg, P.,
Venkataramiah, N., Duda, K., Ganga, I., Lin, G., Pearson,
M., Thaler, P., and C. Tumuluri, "NVGRE: Network
Virtualization using Generic Routing Encapsulation",
draft-sridharan-virtualization-nvgre-03 (work in
progress), August 2013.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The
Locator/ID Separation Protocol (LISP)", RFC 6830, January
2013.

- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.

Authors' Addresses

Fabio Maino
Cisco Systems
170 Tasman Drive
San Jose, California 95134
USA

Email: fmaino@cisco.com

Vina Ermagan
Cisco Systems
170 Tasman Drive
San Jose, California 95134
USA

Email: vermagan@cisco.com

Yves Hertoghs
Cisco Systems
6a De Kleetlaan
Diegem 1831
Belgium

Phone: +32-2778-435
Fax: +32-2704-6000
Email: yves@cisco.com

Dino Farinacci
lispers.net

Email: farinacci@gmail.com

Michael Smith
Insieme Networks
California
USA

Email: michsmit@insiemenetworks.com